Journal of Energy Storage 77 (2024) 109884

Contents lists available at ScienceDirect

Journal of Energy Storage

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

Research papers

Open access dataset, code library and benchmarking deep learning approaches for state-of-health estimation of lithium-ion batteries

Fujin Wang, Zhi Zhai, Bingchen Liu, Shiyu Zheng, Zhibin Zhao [*], Xuefeng Chen

National Key Lab of Aerospace Power System and Plasma Technology, Xi'an Jiaotong University, Xi'an, 710049, PR China
School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an, 710049, PR China

ARTICLE INFO

Keywords:
Lithium-ion battery
State-of-health (SOH) estimation
Deep learning
Benchmark study
Open-source codes
Lithium-ion battery dataset

ABSTRACT

Great progress has been made in deep learning (DL) based state-of-health (SOH) estimation of lithium-ion batteries, which helps to provide recommendations for predictive maintenance and replacement of lithium-ion batteries. However, despite the abundance of articles, few open-source codes are publicly available. While there are several public datasets, they tend to be more oriented toward simulating laboratory environments rather than real-world usage scenarios. Moreover, they solely provide raw data without any corresponding preprocessing codes, resulting in inconsistencies in preprocessing methods across different papers. These reasons lead to unfair comparisons and ineffective improvements. In response to these problems, this paper publishes a large-scale lithium-ion battery run-to-failure dataset, consisting of 55 batteries, and provides a unified data preprocessing method. Besides, we comprehensively evaluate 5 well-known DL-based models to provide benchmark research. To be specific, first, the existing DL-based SOH estimation methods are reviewed in detail. Second, we provide a comprehensive evaluation of DL-based models on 2 large-scale datasets, including 100 batteries, with 3 input types and 3 normalization methods. Third, we make the complete evaluation codes and dataset publicly available for better comparison and model improvement. Fourth, we discuss future DL-based SOH estimation, including unsupervised learning, transfer learning, interpretability, and physics-informed machine learning. We emphasize the importance of open-source code, provide baseline estimation errors (error upper bounds), and discuss existing issues in this field. The code library is available at: https://github.com/wang-fujin/SOHbenchmark.

1. Introduction

Lithium-ion batteries have the advantages of high energy density, low self-discharge rate, and long lifetime [1]. As one of the most widely used energy storage devices in modern society, lithium-ion batteries played an indispensable role in portable rechargeable devices [2], electric vehicles [3,4], energy storage power stations [5], satellites [6], and other fields [7]. The performance of lithium-ion batteries degrades due to the degradation of their electrochemical components and the increase of internal resistance, resulting in capacity and power decay, also known as battery aging. The aging and damage of lithium-ion batteries may lead to the failure of the power system, causing property damage and personal injury [8]. Therefore, the prognostics and health management (PHM) of lithium-ion batteries are very important. It is essential to estimate battery health, and model battery degradation aging to meet the required operational performance and optimize the usage process [9].

State-of-health (SOH) and remaining useful life (RUL) are the most important parameters to evaluate the current battery health and performance [10]. SOH is defined as the ratio of the maximum available capacity of the current cycle to the initial capacity [11]. RUL is closely related to SOH and is defined as the number of cycles remaining from the current cycle until the SOH drops to 80%. SOH can track the actual performance of the battery in the application, describing the degree of degradation and aging. Unfortunately, SOH cannot be directly measured. Generally speaking, only several variables such as voltage, current, and temperature can be directly measured. SOH can be estimated from these measurements. Thus, how to estimate SOH more accurately and effectively is still a challenging problem.

In recent years, various SOH estimation methods of lithium-ion batteries have been proposed, which greatly advance the development of this field. The SOH estimation of lithium-ion batteries aims to evaluate the current state of the battery and provide recommendations for predictive maintenance and replacement, thereby improving

[*] Corresponding author at: School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an, 710049, PR China.
E-mail address: zhaozhibin@xjtu.edu.cn (Z. Zhao).

https://doi.org/10.1016/j.est.2023.109884
Received 27 July 2023; Received in revised form 23 October 2023; Accepted 23 November 2023
Available online 9 December 2023

the reliability and safety of the system operation. Approaches for SOH estimation typically include model-based methods and data-driven methods. Model-based methods, such as electrochemical models [12], equivalent circuit models [13], etc., usually require the help of expert knowledge and experience to determine the model parameters. In contrast, the data-driven methods [14] require minimal knowledge of battery aging mechanisms and can mine representative information from historical data.

Deep learning (DL), as a powerful data mining technology and one of the representatives of data-driven methods, has been widely used in various fields and achieved great success, including computer vision [15,16], natural language processing [17,18], automatic driving [19,20], mechanical fault diagnosis [21–23], etc. In the field of battery health management, many scholars have applied DL technology, including multi-layer perception (MLP), convolutional neural network (CNN), recurrent neural network (RNN), Transformer, and their various variants, to battery SOH estimation [24–26], since the public availability of some large battery datasets. However, models proposed by different researchers often have unique input types and hyper-parameters. Publicly available large battery datasets also do not provide a unified data preprocessing method, resulting in different preprocessing methods in different papers. Unfortunately, few scholars have made their codes public, leading to unfair comparisons and ineffective improvements of methods proposed in papers. To address this problem, it is crucial to provide a benchmark study and open-source codes. In this uniform benchmark, the researchers can make a fair comparison and propose more appropriate and efficient methods.

For a comprehensive comparison, it is necessary to utilize different datasets and compare the performance of different models on a unified platform. What is more, different data preprocessing methods and input types also have a great impact on the model performance. Without controlling for variables, it would be difficult to discern whether improvements in results stem from enhancements to the model itself or modifications to data preprocessing methods.

To address these issues, we conduct experiments on two large datasets, with a total of 100 batteries. One is the Toyota-MIT-Stanford dataset [27], which contains 45 batteries, and the other is a new dataset we will make publicly available, which contains 55 batteries. Specifically, we compare and evaluate the performance of DL-based models in the lithium-ion battery SOH estimation task in detail from different perspectives, including different DL models, input types, and normalization methods. Based on this benchmark study, we evaluate and discuss the results, and explore future research directions for SOH estimation tasks of lithium-ion batteries. We also release a code library to evaluate DL-based SOH estimation models and provide a baseline, the upper bound of the model's estimation error, to avoid ineffective improvements. Through these works, we aim to provide a fair comparison environment and discuss the issues of existing methods. We also emphasize the importance of open-source codes. Although many review papers have been published on battery health management, to the best of our knowledge, this is the first work to conduct a comprehensive benchmark study and release the code library to the public. The benchmark code library is available at: https://github.com/wang-fujin/SOHbenchmark.

The overall organizational structure of our paper is shown in Fig. 1. Our contributions are as follows:

1. We develop a large battery dataset, which contains run-to-failure data of 55 batteries under 6 charging and discharging strategies. In particular, there are some of our unique charging and discharging strategies. For example, we simulate the charging and discharging strategies of satellites in geosynchronous earth orbit (GEO), which have not been seen in other public datasets.
2. We evaluate various vanilla DL algorithms and data preprocessing methods for SOH estimation of lithium-ion battery to provide a benchmark. A total of 5 DL models are implemented. Each model is evaluated on 100 batteries with different preprocessing, including 3 input types and 3 normalization methods.
3. We release the code library for the better comparisons. It is a unified battery SOH estimation library, which retains an extended interface for everyone to load their own datasets and models to carry out new studies.

The remainder of this paper is organized as follows. In Section 2, we give a review of the development of DL methods in the field of lithium battery SOH estimation. Section 3 to Section 7 present the evaluation algorithm, datasets, data preprocessing, evaluation methodologies, and evaluation results, respectively. Section 8 discusses the evaluation results and further discusses future research directions. The conclusions are given in Section 9.

2. Review

2.1. Review screening methods

In recent years, DL-based methods have been widely used in the field of battery health management and prognosis. This paper mainly focuses on a benchmark study of deep learning in SOH estimation of lithium-ion batteries, excluding SOC estimation and RUL prediction. The ScienceDirect, IEEE Xplore, and Springer were used to find relevant papers for this benchmark study. We used joint keywords, such as *("li-ion battery" OR "lithium-ion battery" OR "lithium battery") AND ("capacity" OR "state of health") AND "deep learning"*, for our initial search. More than 700 literature were searched. We limited the publication year of the article to 2017 to the present (early 2023), limited the article type to journal articles, and conducted a brief analysis of the article titles. A total of nearly 300 articles have been downloaded. Finally, we conducted detailed screening based on abstracts and paper content, and selected about 100 papers from the download list for review in this paper.

2.2. Literature reviews

Many scholars have done some work on the SOH prediction of lithium-ion battery. According to the literature retrieved above, we make a brief review and summary in this part. A summary and classification of DL-based SOH estimation methods are given in Table 1.

2.2.1. RNN-based model

Recurrent Neural Network (RNN) is often used to deal with problems where sequence data is taken as input. Based on vanilla RNN, many variants have been developed, such as long short-term memory (LSTM) network and gate recurrent unit (GRU). This part will review the work of the RNN-based model in SOH estimation problem.

In Che's research [28], an LSTM model using probabilistic regression was applied to predict battery capacity, which extracted health indicators (HIs) from partial capacity and voltage series as input. The similar studies were conducted by [29,30], and they established a health assessment framework based on LSTM. To solve the problem that laboratory feature extraction methods are inapplicable due to highly unstable operating conditions of electric vehicles, Heinrich et al. [31,32] used LSTM to model the electric response during capacity testing and proposed a SOH estimation method based on virtual battery experiments. Aiming at the problem that hyper-parameters of LSTM are difficult to pre-define in the SOH estimation problem, optimization algorithms such as particle swarm optimization (PSO) [33] and cuckoo search [34] were used to optimize hyper-parameters to speed up the convergence of the model. Chen et al. [35] used empirical mode decomposition (EMD) to extract features from the battery data and then fed the features into different deep RNNs to estimate the SOH. To address the problem of complex calculation of incremental capacity (IC) curve, Zhang et al. [36] improved the IC curve calculation method and established an SOH estimation model based on LSTM using IC features as input. Using features as input to estimate SOH is a

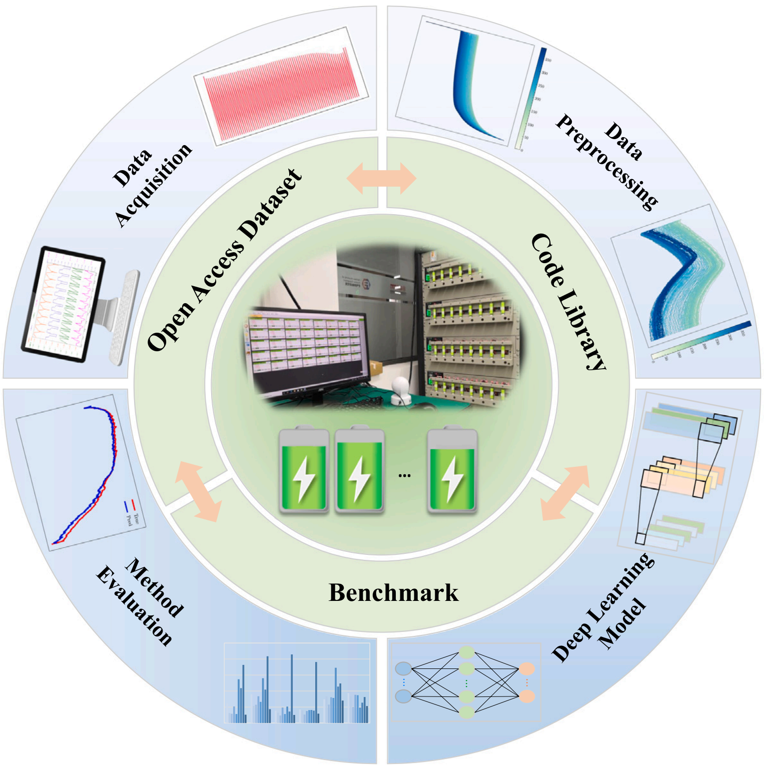


Fig. 1. The organizational structure of this paper.

popular approach [37,38]. Researchers have designed various feature extraction methods to obtain more representative features to improve the estimation accuracy of SOH . Chen [39], Li [40], and Tian [41] et al. extracted features from charging data as input to the model to predict SOH. In contrast, some studies [42,43] extracted features from discharge data. Ma et al. [44,45] proposed a method for battery SOH prediction based on LSTM network, which improved the prediction accuracy by extracting features from the differential thermal voltammetry (DTV) curve. However, all the above studies are based on feature engineering, and the performance of the model heavily depends on the expert experience and the quality of handcraft features.

Considering the powerful feature extraction ability of DL, many studies directly use raw data as input to estimate SOH. Li et al. [46, 47] proposed a LSTM-based model which can predict battery health without feature extraction. What is more, Refs. [48,49] used raw charging and discharging data as model input to predict capacity and SOH. Wang et al. [50] proposed a Bi-LSTM model, using normalized capacity time series data as the training data of the model, and transforming the RUL problem into a time series prediction problem [51]. Li et al. [52] proposed a framework for capacity prediction through multitasking learning which uses the RNN-based sequence-to-sequence model. Considering the volatility and non-linearity in battery degradation, Cheng [53] and Chen [54] combined EMD and LSTM to predict SOH and degradation trajectories. Zhang et al. [55] designed a model that can update the hyper-parameters based on historical data, which can better track the capacity degradation trajectory. Hong et al. [56] proposed a method for online SOH estimation of electric vehicle batteries using variable-length input LSTM networks, which can learn battery degradation factors according to different driving phases.

The model with raw data as input can perform an end-to-end SOH prediction, omitting the time-consuming and laborious feature extraction process. However, the performance of these models relies more on their feature extraction capabilities than those based on feature engineering, which requires the model to have a strong feature extraction capability to predict SOH more accurately.

2.2.2. CNN-based model

Convolutional neural networks (CNN) have achieved great success in computer vision. In the field of battery SOH estimation, a large number of CNN-based methods have also been proposed. Yang et al. [57] completed feature extraction of two consecutive charge and discharge cycles by CNN, and then predicted SOH through a random forest algorithm. Ruan et al. [58] proposed a CNN-based method to automatically extract degradation features from reliable ΔQ signals. To achieve online forecasting, Shen et al. [59] proposed a model that utilizes deep neural network with multiple convolutional layers to estimate lithium-ion battery capacity online. Their method has a higher accuracy compared to the traditional machine learning methods in online prediction.

In the actual industry, current, voltage, and temperature are often easy to collect, so most methods estimate SOH based on these three measurements. However, there are also methods to improve prediction accuracy via exploring more variables. Xiong et al. [60] used the raw impedance spectrum as the input quantity of CNN. They constructed an input reconstruction module to maintain high accuracy of battery capacity prediction even with a large number of unlabeled samples. Pradyumna et al. [61] proposed a capacity estimation method based on electrochemical impedance spectroscopy (EIS), but this method could only predict off-line and accurately only at 100% SOC. Some scholars also consider from the perspective of the model input, and try to improve the model performance via changing the series data into other forms. Costa et al. [62] applied dynamic time warping (DTW) technology to convert battery data into images, which are inputted into CNN to complete the characterization of a battery degradation process. Sohn et al. [63] proposed a method to monitor battery degradation by online predicting the knee-point of battery capacity. Similarly, Ji et al. [64] also proposed a method for SOH estimation combined with knee-point probability prediction.

Considering the incomplete charging process, several papers use partial data to predict SOH [65,66]. Fan et al. [67] proposed a method to estimate battery capacity using voltage data from the first 10 s of a battery discharge curve. Deng et al. [68] proposed three data-driven methods to estimate SOH based on short random charging segments. The methods have been verified on commercial lithium-ion batteries at different temperatures and discharge rates, but require constant temperature, discharge current, and complete charge/discharge cycles. Similarly, Tian et al. [69,70] proposed a DL method based on CNN, using a small portion of charging data to predict battery degradation. Saxena et al. [71] designed a CNN model to predict the entire battery capacity decay curve using data from the first 100 cycles.

To account for temporal information in battery data, Zhou et al. [72] proposed a Temporal-Convolutional Network (TCN) based battery SOH estimation model that uses voltage and temperature degradation trends as inputs and takes health characteristics into account. In Bockrath's research [73], TCN is also applied to estimate the battery health state.

CNN has a weight-sharing property that allows it to accommodate more layers with the same number of parameters in comparison to other models. Additionally, its superior feature extraction ability has been tested and proven in the realm of computer vision. In battery SOH prediction, CNN is also widely used for feature extraction and has achieved great results.

2.2.3. Other DL-based models

In addition to RNN-based and CNN-based models, some other DL-based models are also used for battery SOH estimation. Due to its relatively small number of articles, we will focus our review on this part.

Wang et al. [74] utilized a dual self-attention mechanism to build the network, and used local and global self-attention modules to enhance the ability of SOH prediction. Qian et al. [75] combined historical state information and future load information to proposed an attention-based sequence-to-sequence model consisting of two encoders and one decoder for SOH prediction. To modeling battery degradation timing, Wang et al. [76,77] proposed a degradation trend alignment method based on cycle-consistency learning to align battery degradation data, and then got the health state based on the aligned features. Kim et al. [78] employed information-maximizing generative adversarial networks (GAN) to identify potential features, followed by using Gaussian process regression to predict battery capacity. This method can predict the discharge capacity of the battery well. To estimate the capacity of lithium-ion batteries according to the time series characteristics in the process of battery degradation, Cui et al. [79] proposed a model using feature transformation process neural network. Mckay et al. [80] proposed a method that can generate data based on an electrochemical model and then predict lithium-ion battery performance based on MLP. In addition, Autoencoder (AE) [81,82], MLP [83,84], deep belief network (DBN) [85,86], radial basis function (RBF) network [87], extreme learning machine (ELM) [88,89], etc., are also used for the task of SOH prediction. Xue et al. [90] adopted generalized regression neural network to complete the construction of SOH prediction model, and used quantum genetic algorithm (QGA) to optimize the hyper-parameters of the network. Qaisar et al. [91] put forward an event-based approach to extract and forecast battery capacity, highlighting the benefits of utilizing artificial neural networks (ANNs).

Scholars seem to be interested in using handcraft features as input, regardless of the type of network being used. Song et al. [92] designed a feedforward neural network (FFNN) with handcraft features as input to estimate SOH. In [93], Maleki proposed a DL structure combining autoregressive integrated moving average (ARIMA) and knowledge transfer asset to achieve the estimation of SOH. The least absolute shrinkage and selection operator (LASSO) regression and Pearson correlation coefficient are used to selecting the four most influential features as the input of the network. Lin et al. [94] adopted temperature and features extracted from voltage signals as inputs to the model which is an improved fuzzy cerebellar model neural network (IFCMNN) designed to estimate SOH and has good universality.

2.2.4. Hybrid models

The hybrid model refers to a model that combines various types of basic networks to improve the accuracy of SOH prediction. For example, Pepe et al. [95] proposed a battery health prediction model incorporating neural networks and ordinary differential equation (ODE). Zhang et al. [96] proposed a battery state prediction method based on GAN-CLS and Bi-LSTM. It uses GAN-CLS to generate data and Bi-LSTM to train a model. Yao et al. [97] used three neural networks, CNN, LSTM, and graph neural network (GNN), to analyze the 27 HIs extracted from the charging and discharging data. It is interesting to note that many scholars [98–102] choose CNN to extract hidden information from data, and utilize LSTM to process time series information for enhancing the model performance. In addition, combining LSTM or CNN with attention mechanism is also a common method [103–107], in which the attention mechanism is used to extract more robust features.

Although the hybrid model may have higher predictive accuracy, its model architecture is more complex and consists of a larger number of parameters. It is worth investigating whether combining multiple networks is necessary for enhancing model performance in estimating SOH. This investigation would also entail examining the individual performance of each fundamental network structure in SOH estimation.

2.3. A brief review of reviews

In recent years, due to the efforts of scholars, the research on battery health management has made great progress. An interesting observation is that many review papers related to this topic have also been published. We reviewed the application of DL in battery SOH estimation above. Here, we make a summary of some review papers, so that readers can quickly find the articles they want to know.

Refs. [108–110] reviews the current status of SOC estimation, classifies SOC estimation methods, and discusses the advantages, disadvantages, limitations, and future research directions of the algorithms. Similarly, Ref. [111–114] mainly reviews the current state of SOH estimation. Some articles provide an overall review of battery PHM [24,115,116], including SOC, SOH, and RUL prediction.

Che et al. [117] reviewed the lithium battery aging mechanism and the latest health prediction methods, and summarized the main challenges and research prospects of battery health prediction. Li et al. [14] discussed the strengths and limitations of data-driven approaches for lithium battery health estimation and lifetime prediction, and further discussed the challenges of battery health management and potential next-generation technologies. Zhang et al. [11] assessed the PHM of lithium-ion batteries based on DL-based method, introduced commonly used data, methods, and evaluation metrics, and looked forward to the prospect of using DL technology to achieve effective PHM of lithium-ion batteries. Similarly, Lipu [118] also conducted a survey on the application of DL-based method in SOC, SOH, and RUL prediction. He mainly focused on methods, implementations, strengths, weaknesses, contributions, etc., and also explored various limitations and challenges of DL in BMS related to battery, algorithmic, and operational issues.

Although many reviews have been published, they did not provide a benchmark for comparison by subsequent scholars. In this paper, we not only review the existing DL-based SOH estimation methods, but also publish a large dataset and provide a code library to facilitate other scholars to study and compare their individual methods.

3. Evaluation algorithms

It is impossible to cover all the published models since there is currently no open source community in the field of battery SOH estimation. Therefore, we evaluate 5 well-known DL-based models, including MLP, CNN, LSTM, GRU, and Attention model. It is worth noting that we only evaluate they vanilla structures without any other variants. In this section, we briefly explain these five algorithms.

Table 1
DL-based SOH estimation methods.

Reference	Method	Dataset	Input type	Number of features
[28]	LSTM	MIT-Stanford	Features from charging data	6
[29]	LSTM	NASA	Features from charging and discharging data	
[33]	LSTM	Data from the laboratory	Features from charging and discharging data	4
[34]	DACS-LSTM	NASA, CALCE	Features from charging and discharging data	9
[35]	LSTM	NASA	Features from charging and discharging data	4
[37]	GRU-HMM	NASA, Oxford	Features from charging and discharging data	3
[38]	Bi-LSTM	NASA	Features from charging data	
[39]	GRU	Oxford	Features from charging data	6
[40]	LSTM, BiLSTM	NASA, Lishen LR1865SZ	Features from charging data	1
[42]	KIRNN-Monte Carlo	Data from the laboratory	Features from discharging data	10
[43]	GPR, LSTM, ANN	NASA, CALCE, Oxford	Features from discharging data	5
[44]	LSTM	NASA	Features from charging and discharging data	3
[46]	LSTM	RWTH	Charging and discharging data	
[47]	LSTM	RWTH	Charging data	
[48]	LSTM	NASA, CALCE	Charging and discharging data	
[49]	AST-LSTM	NASA	Charging and discharging data	
[52]	Sequence to Sequence	RWTH	Charging and discharging data	
[53]	LSTM-EMD	CALCE	Charging data	
[55]	LSTM	NASA	Charging and discharging data	
[56]	LSTM	EVs dataset in Beijing	Charging data	
[57]	CNN-RF	MIT-Stanford	Features from charging and discharging data	
[58]	CNN	Oxford	Charging and discharging data	
[59]	Deep CNN	NASA	Charging data	
[60]	CNN	Data from the laboratory	Charging and discharging data	
[61]	CNN	Data from the laboratory	Charging and discharging data	
[62]	DTW-CNN	Data from the laboratory	Charging and discharging data	
[63]	CNN	MIT-Stanford	Charging and discharging data	
[67]	CNN	Oxford	Discharging data	
[69] [70]	CNN	NASA, CALCE, Oxford	Charging data	
[72]	Attention depthwise-TCN	NASA	Charging data	
[73]	TCN	NASA	Charging and discharging data	
[74]	Dual self-attention	Data from the laboratory	Features from charging data	15
[78]	EISGAN, GPR	Data from the laboratory	Charging and discharging data	
[81]	Encoder–decoder	NASA	Charging and discharging data	
[83]	MLP	NASA	Features from charging and discharging data	5
[84]	MLP	NASA	Features from discharging data	6
[85]	DBN	NASA	Features from discharging data	17
[87]	RBF-AR	NASA, CALCE	Features from charging and discharging data	3
[88]	ELM	Oxford	Features from charging and discharging data	5
[89]	ISSA-DELM	NASA	Features from discharging data	2
[90]	PF-QGA-GRNN	NASA, real vehicle dataset	Charging and discharging data	
[91]	ANN	NASA	Features from discharging data	3
[95]	NN based on ODE	NASA, Oxford	Features from charging data	1
[97]	CNN, LSTM, GNN	MIT-Stanford	Features from charging and discharging data	27
[99]	CNN-LSTM	NASA	Charging and discharging data	
[103]	QGA-ASM-LSTM	NASA, CALCE	Features from charging and discharging data	28
[104]	LSTM, Attention, GCN	NASA	Features from charging and discharging data	8
[107]	CNN-transform	NASA	Features from charging and discharging data	4
[119]	SSL	CALCE, Oxford	Features from charging data	3

Note: NASA dataset comes from [120], MIT-Stanford dataset comes from [27,121], Oxford dataset comes from [122], CALCE dataset comes from [123], and NWTH dataset comes from [46].

3.1. MLP

MLP gained popularity in the machine learning field in the 1980s [124]. The MLP has a forward structure where each layer is fully connected, as shown in Fig. 2. A MLP consists of several hidden layers, an input layer, and an output layer. MLP can be calculated using the following formula:

$$y = \phi(W \cdot x + b), \tag{1}$$

where $\phi(\cdot)$ represents the activation function, W is the weight, b denotes bias, and x is the input.

MLP can be used to solve the simple prediction and classification problems. However, as the complexity of the task increases, the MLP's computational load is too heavy, resulting in unsatisfactory performance for some time series problem.

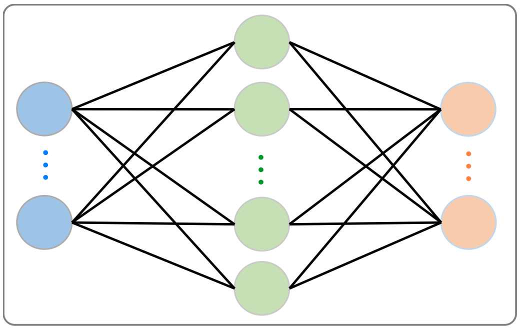


Fig. 2. The diagram of MLP.

3.2. CNN

In 1980, the neurocognitive machine model was developed, which is the first neural network model with the convolutional structure [125]. A dozen years later, LeCun et al. [126] designed the LeNet-5 network in 1998, which applied CNN to handwritten digit recognition and trained it using the backpropagation algorithm. In 2012, Krizhevsky

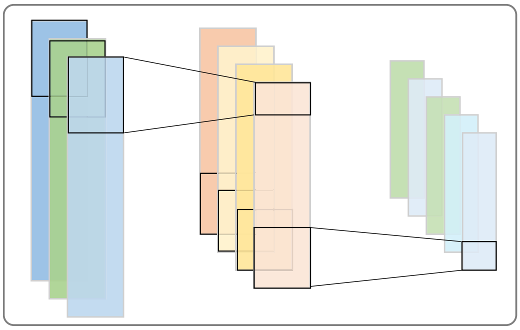


Fig. 3. The diagram of CNN.

et al. [15] achieved a significant breakthrough in the ImageNet image classification competition by using the AlexNet. Besides, He et al. [127] have won the champion of ILSVRC in 2015, and ResNet has become a milestone in the history of CNN.

Generally, CNN is composed of convolutional layer, activation function, pooling layer, and fully connected layer, as shown in Fig. 3.

For an input x, the convolutional layer can be defined as a multiplication with a filter kernel w, the feature map after the convolutional layer can be formulated as:

$$h_k^l = \phi(w_k^l \star x + b_k^l), \tag{2}$$

where $\star$ denotes the convolution operator, h_k^l, w_k^l, and b_k^l denote the obtained feature map, the weight, and the bias of the kth convolutional kernel of the lth layer, respectively.

Activation functions are used to add nonlinear factors, improve the expression ability of neural networks, and solve problems that cannot be solved by linear models. ReLU is a common activation function, it can be described as:

$$\mathrm{ReLU}(h_k^l) = \max(0, h_k^l), \tag{3}$$

where h_k^l denotes the output of the convolution layer.

The pooling layer is set behind the convolutional layer to downsample the feature map, thereby reducing the parameters and computation. Common pooling operations include Max Pooling and Average Pooling. The maxpooling layer can be defined as:

$$z_k^l = \mathrm{down}(\mathrm{ReLU}(h_k^l); s), \tag{4}$$

where $\mathrm{down}(\cdot)$ denotes the down-sampling function of the maxpooling layer, z_k^l and s denote the output feature map of the pooling layer and pooling size, respectively.

After a series of convolutional and pooling layers, the high-level features of the input data are extracted and fed into the fully connected layer, which is used to convert the high-dimensional feature vector into a low-dimensional vector and output the final result.

3.3. LSTM

LSTM, proposed in 1997 [128], is a variant of RNN, which has better processing capability for gradient disappearance in long sequence data training. The structure of LSTM is shown in Fig. 4.

LSTM has two transitive states, c_t and h_t, which improve the processing efficiency compared to RNN's one state. An LSTM cell contains a forget gate f_t, an input gate i_t, and an output gate o_t. The calculation process of LSTM can be expressed as:

$$f_t = \sigma\left(W_f \cdot \left[h_{t-1}, x_t\right] + b_f\right), \tag{5}$$

$$i_t = \sigma\left(W_i \cdot \left[h_{t-1}, x_t\right] + b_i\right), \tag{6}$$

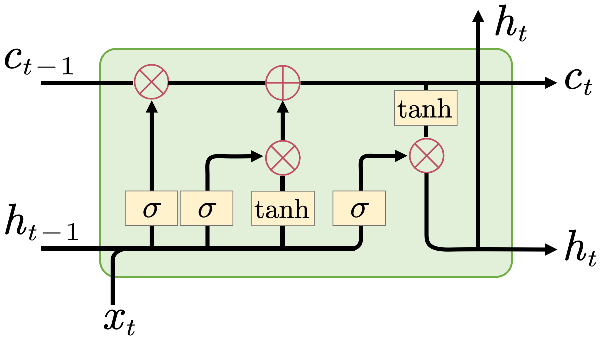


Fig. 4. The diagram of LSTM.

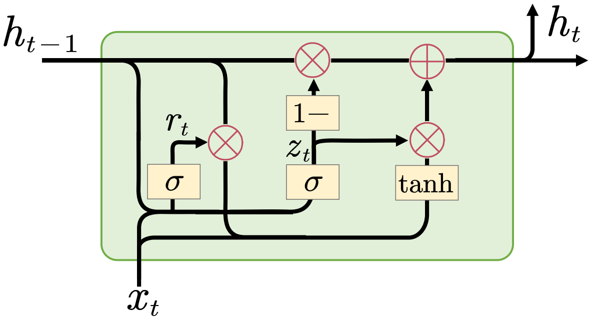


Fig. 5. The diagram of GRU.

$$\tilde{c}_t = \tanh\left(W_c \cdot \left[h_{t-1}, x_t\right] + b_c\right), \tag{7}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \tag{8}$$

$$o_t = \sigma\left(W_o \cdot \left[h_{t-1}, x_t\right] + b_o\right), \tag{9}$$

$$h_t = o_t \odot \tanh\left(c_t\right). \tag{10}$$

where $\sigma(\cdot)$ represents the Sigmoid function, W_f, W_i, W_c, and W_o represent weighted matrix, b_f, b_i, b_c, and b_o represent bias, and $\odot$ stands for Hadamard Product, which is the multiplication of the corresponding entries in the matrices.

3.4. GRU

GRU, proposed in 2014 [129], is also a kind of RNN. It has good applicability to the backpropagation gradient problem and time series problem. GRU has a lower computational cost compared to LSTM, since it is simpler to compute and train. The overall structure of GRU is shown in Fig. 5.

The update gate z_t of GRU is composed of the input gate in LSTM merged with the forget gate, and the reset gate r_t is composed of the hidden layer and memory unit of LSTM. The formulas are as follows:

$$z_t = \sigma\left(W_z \cdot \left[h_{t-1}, x_t\right] + b_z\right), \tag{11}$$

$$r_t = \sigma\left(W_r \cdot \left[h_{t-1}, x_t\right] + b_r\right), \tag{12}$$

$$\tilde{h}_t = \tanh\left(W_h \cdot \left[r_t \odot h_{t-1}, x_t\right] + b_h\right), \tag{13}$$

$$h_t = \left(1 - z_t\right) \odot h_{t-1} + z_t \odot \tilde{h}_t. \tag{14}$$

where W_z, W_r, and W_h represent weighted matrix, b_z, b_r, and b_h represent bias, and $\tilde{h}_t$ is hidden state.

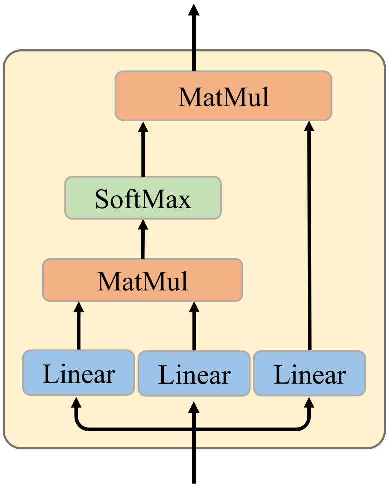


Fig. 6. The diagram of self-attention.

By incorporating a gating mechanism for selective memory and forgetting, this design simplifies the structure and improves computational performance.

3.5. Attention

Attention is a technique that allows a model to focus on the key parts of its input or output. It was originally a concept used in the visual domain of image classification. It was later widely used in natural language processing, especially in machine translation. In 2014, the Google Mind team applied Attention mechanism to recurrent models of RNN to classify images [130]. Bahdanau et al. [131] applied Attention mechanism to machine translation in 2015. The simultaneous translation and alignment of a machine translation task using an attention-like mechanism is the first application of Attention on neural language processing. Besides, the Google Machine translation team made extensive use of self-attention methods to learn text representation in 2017 [132].

Take Self-Attention for example. The original formula for Attention can be expressed as:

$$\mathrm{Attention}(Q,K,V)=\mathrm{softmax}(\frac{QK^T}{\sqrt{d_k}})V, \tag{15}$$

where Q is analogous to inquiry, K is analogous to index, V is analogous to an answer, and d_k is the dimension of the hidden layer. Q and K obtain a mask matrix with all values of 0–1, and V is used to preserve the input characteristics. The diagram of Self-Attention is shown in Fig. 6.

4. Datasets

In this paper, we conduct the experiments on two large datasets, with a total of 100 batteries, for the battery SOH estimation. We provide the detailed experimental results for each battery.

4.1. XJTU battery dataset

The XJTU battery dataset is a new large-scale dataset we designed for this benchmark. There are 55 lithium-ion batteries subjected to 6 charging and discharging strategies in this dataset, which records their run-to-failure data. These lithium-ion batteries were manufactured by LISHEN, whose chemical composition is $\mathrm{LiNi_{0.5}Co_{0.2}Mn_{0.3}O_2}$. They have a nominal capacity of 2000 mAh, a nominal voltage of 3.6 V, a charge cut-off voltage of 4.2 V, and a discharge cut-off voltage of 2.5 V. All batteries cycled to failure in 40-channel ACTS-5V10A-GGS-D at room temperature, as shown in Fig. 7. We use *batch 1* to *batch 6* to represent the 6 charging and discharging strategies, respectively. All batches except batch 2 consist of 8 batteries, while batch 2 contains 15 batteries. All battery data are available at: https://wang-fujin.github.io/.

4.1.1. Batch 1

The batteries in batch 1 were cycled under a fixed charging and discharging strategy. All batteries were charged to 4.2 V at 2 C with constant voltage and constant current (CC–CV) mode and then discharged to 2.5 V at 1 C.

4.1.2. Batch 2

Batch 2 contains 15 batteries, and its charging and discharging strategy is similar to batch 1. All batteries were charged to 4.2 V at 3 C with CC–CV mode and then discharged to 2.5 V at 1 C.

4.1.3. Batch 3

Batch 3 has a more complex protocol than that of the first two batches. All batteries were charged at 2 C with CC–CV mode. Then they were discharged to 2.5 V with a current value of x C, where $x \in \{0.5, 1, 2, 3, 5\}$.

4.1.4. Batch 4

Batch 4 is similar to batch 3. The batteries were charged at 2 C with CC–CV mode and then discharged to 3.0 V with the same current as batch 3.

4.1.5. Batch 5

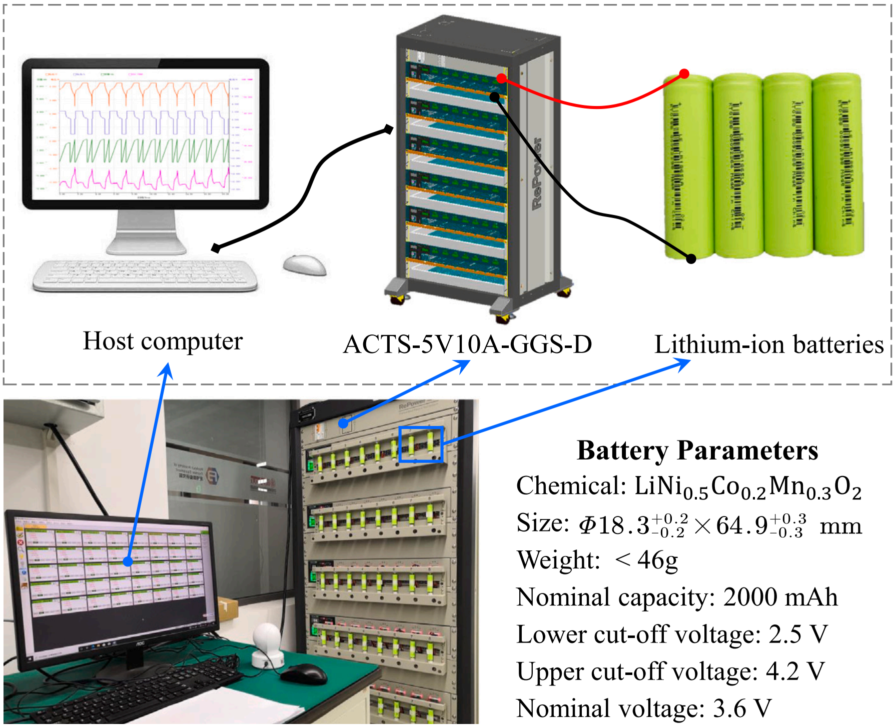
Batch 5 follows the random walking strategy, thereby the entire process of charging and discharging are more closely with real-life usage. Specifically, all cells are charged to 4.2 V at 1 C with CC–CV mode and then discharge to 3.0 V. The discharge current is a random integer in the range of $[2, 8]$ ampere and the duration is in the range of $[2, 6]$ min.

4.1.6. Batch 6

In batch 6, we simulated the charging and discharging strategy of a satellite in geosynchronous earth orbit (GEO). The batteries of GEO satellites only supply power during the shadow period of the earth, and the depth-of-discharge (DOD) is generally less than 80% [133]. The duration of each discharge is determined by the duration of the Earth's shadow. Therefore, the discharge duration of each cycle is different, and the DOD is also different. More details can be found in Appendix A.

4.2. Toyota-MIT-Stanford dataset

The Toyota-MIT-Stanford dataset [27] contains data of LiFePO4 (LFP) batteries cycled under fast-charging. These lithium-ion cells were manufactured by A123 Systems (APR18650M1A). They have a nominal capacity of 1.1 Ah and a nominal voltage of 3.3 V. In fast-charging experiments, these batteries were charged to 80% SOC through 4 stages of fast-charging in constant current (CC) mode and then fully charged in a CC–CV model. During the first three stages, each stage (0%–20%, 20%–40%, and 40%–60%, respectively) charges 20% SOC at a predetermined rate. In the fourth stage, the charging rate is determined by ensuring that the total charge time of four stages is 10 min. Then all batteries were discharged at 4 C in a CC mode. The run-to-failure data of 45 batteries from the last batch were used in this paper. Since these batteries contain 9 charging strategies, we divide them into 9 groups, labeled as *group 1* to *group 9* respectively in Table 2.



Battery Parameters

Chemical: $LiNi_{0.5}Co_{0.2}Mn_{0.3}O_2$

Size: $\Phi 18.3^{+0.2}_{-0.2} \times 64.9^{+0.3}_{-0.3}$ mm

Weight: < 46g

Nominal capacity: 2000 mAh

Lower cut-off voltage: 2.5 V

Upper cut-off voltage: 4.2 V

Nominal voltage: 3.6 V

Fig. 7. The schematic diagram of the experimental platform.

Table 2
Details about the Toyota-MIT-Stanford dataset.

Group	Charge protocol 0-20%-40%-60%-80%	Discharge protocol	Channel
1	3.6-6-5.6-4.755	4C	11,12,27,29,38
2	4.4-5.6-5.2-4.252		8,15,18,32,48
3	4.8-5.2-5.2-4.16		1,2,10,20,42
4	5.2-5.2-4.8-4.16		6,7,37,41,45
5	6-5.6-4.4-3.834		9,21,22,31,36
6	7-4.8-4.8-3.652		3,25,26,28,44
7	8-4.4-4.4-3.94		13,16,23,24,47
8	8-6-4.8-3		14,17,30,35,39
9	8-7-5.2-2.68		19,33,34,40,43

5. Data preprocessing

The data preprocessing methods, including the type of input data and the normalization way, have a great influence on the performance of the model. In this paper, we compared the impact of three input types on model performance. Meanwhile, each input type is normalized by three methods.

5.1. Input type

From the articles reviewed above, it can be found that the input types of the models proposed in different papers are not uniform, which makes it hard to form a fair comparison among models. To provide a benchmark, we evaluated the impact of three input types on model performance.

It is worth noting that all the three input types come from charging data. Some papers rely on discharging data to estimate SOH, but we think that is unrealistic due to the fact that the battery discharging process is user-specific. That is, different batteries, or even one battery in two discharging cycles, have different discharging strategies. On the contrary, batteries of the same type usually have same charging strategy. Therefore, we believe that it is more realistic to estimate SOH with charging data.

5.1.1. Complete charging data

Using complete charging data as input to the model requires only the simplest preprocessing. First, the charging data of each cycle is divided, and then all charging data are resampled to a fixed length since the number of points sampled in each cycle is different. In this paper, the fixed length is set to 128. To be specific, the time, voltage, current, and temperature data in each cycle are resampled at equal time intervals to 128 points, i.e., the shape of each sample $x_i \in \mathbb{R}^{128\times4}$.

5.1.2. Partial charging data

In some practical industrial applications, batteries are not always fully charged and discharged. Therefore, the partial charging data is used as an input type. We have intercepted the charging data where the charging voltage is within the specified range, 3.7–4.1 V for the XJTU battery dataset and 3.0–3.59 V for the Toyota-MIT-Stanford dataset. Then, all data are resampled to 128 points. The shape of each sample $x_i \in \mathbb{R}^{128\times4}$. Note that the voltage range here is only selected subjectively and has no special meaning.

5.1.3. Handcraft features from charging data

Extracting handcraft features as model input is another hot research topic. Many scholars focus on extracting more representative features. In this paper, we extracted a total of 67 handcraft features from charging data. The description of each feature can be found in Appendix B. The shape of each sample $x_i \in \mathbb{R}^{67\times1}$. It is worth noting that our handcraft features are simply cleaned due to some errors in the calculation process. Specifically, each feature is judged by using the 3σ criterion, and values out of range are replaced by means of linear interpolation.

5.2. Normalization

In the sampled data, the units and scales of the values in each channel are different. The normalization method is often used to preprocess data to make the training process more stable [21]. The common normalization methods are $[0,1]$ normalization, $[-1,1]$ normalization, z-score normalization, and so on. The following is a brief introduction to these common normalization methods:

5.2.1. $[0,1]$ normalization

$[0,1]$ normalization is a linear transformation of the original data to transform the data to the range of [0, 1], and the formula is as follows:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}, \tag{16}$$

where x represents raw data, x_{norm} represents normalized data, and x_{min} and x_{max} are the minimum and maximum values of the raw data, respectively.

5.2.2. $[-1,1]$ normalization

Similar to $[0,1]$ normalization, the $[-1,1]$ normalization is also a linear transformation that can be implemented by:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \times 2 - 1. \tag{17}$$

The meanings of these parameters are the same as those in Section 5.2.1.

5.2.3. Z-score normalization

The z-score normalization is the result obtained by subtracting the mean value from the original data and dividing it by the standard deviation. The formula is as follows:

$$x_{norm} = \frac{x - \bar{x}}{\sigma_x}, \tag{18}$$

where x is raw data, x_{norm} denotes data after normalization, and $\bar{x}$ and σ_x are mean and standard deviation of the raw data, respectively.

6. Evaluation methodology

6.1. Evaluation metrics

To quantitatively evaluate the performance of the models, three evaluation criteria were used in this paper. That is, mean absolute error (MAE), mean absolute percentage error (MAPE), mean squared error (MSE), and the determination coefficient R^2 are employed to evaluate the error between the estimated SOH and true SOH.

6.1.1. MAE

MAE, which is the absolute value of the difference between the predicted value and the true value and then averaged, reflects the magnitude of the average error value. The calculation formula is as follows:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|y_i - \hat{y}_i\right|, \tag{19}$$

where $\hat{y}_i$ and y_i represent the estimated and true SOH, and N denotes the number of samples.

6.1.2. MAPE

MAPE, which is the average of the relative errors, reflects the size of the errors proportionally. The calculation formula is as follows:

$$\text{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{\hat{y}_i - y_i}{y_i}\right|, \tag{20}$$

6.1.3. MSE

MSE, which is the average sum of the squares of each error, is used to measure the difference between the true value and the predicted value. The calculation formula is as follows:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2. \tag{21}$$

Table 3
The number of model parameters.

Model	CNN	LSTM	GRU	MLP	Attention
Parameter	311 505	209 025	158 849	172545	366 089

6.1.4. Determination coefficient

The determination coefficient R^2 is a valuable evaluation metric used to assess how well a regression model fits the data. It measures the proportion of the variance in the dependent variable that is predictable from the independent variables. A higher R^2 value indicates a better fit. The formulation is as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{N}\left(y_i - \bar{y}\right)^2}, \tag{22}$$

where $\bar{y}$ represents the average SOH of all samples.

6.2. Implementation details

To sum up, the flow chart of battery SOH estimation in this benchmarking study is shown in Fig. 8. The details of model structure and training process will be introduced later.

6.2.1. Model structure

The purpose of this paper is to provide a benchmark for DL research in the field of battery SOH estimation. Therefore, all the models we implement in this paper are vanilla models without any other variants. For example, our CNN model is similar to the classic ResNet [127] model. The structures of all models are given in Figs. C.16 and C.17. To ensure relative fairness, the extracted features of all models will eventually pass through the same predictor, which consists of two Linear layers with a ReLU layer embedded in the middle, as shown in Fig. C.17(f). In addition, we try our best to ensure that the number of model parameters is on the same order of magnitude, as shown in Table 3.

6.2.2. Data splitting method

To ensure the completeness of the experiment, each battery in the same batch/group was used as a test set in turn, and the remaining batteries are randomly divided into a training set and a validation set, as shown in Fig. 9.

6.2.3. Experiment details

All models were optimized by minimizing MSE loss. The Adam optimizer with an initial learning rate of 2e−3 and weight decay of 5e−4 was used in the training phase. The batch size was set to 128, the epoch was set to 100, and early stopping is set to 30. A multi-step learning scheduler was used to adjust the learning rate, where gamma is set to 0.5 and milestones are set to $[30, 70]$.

It is worth noting that to ensure comparability among models, all experiments for all models follow the same set of hyper-parameters. To ensure the comparability among different input types of the same model, all input types are preprocessed into the same shape. Considering that the shape of the complete charging data and the partial charging data are both 128 × 4, the shape of the handcraft features is also transformed into 128 × 4 with a Linear layer.

7. Results

We have conducted a comprehensive set of experiments using 3 different input types and 3 normalization methods, resulting in 9 sets of detailed results for each dataset. Specifically, the XJTU battery dataset yielded a set of experimental results consisting of 55 batteries per set, while the Toyota-MIT-Stanford dataset produced a set of results

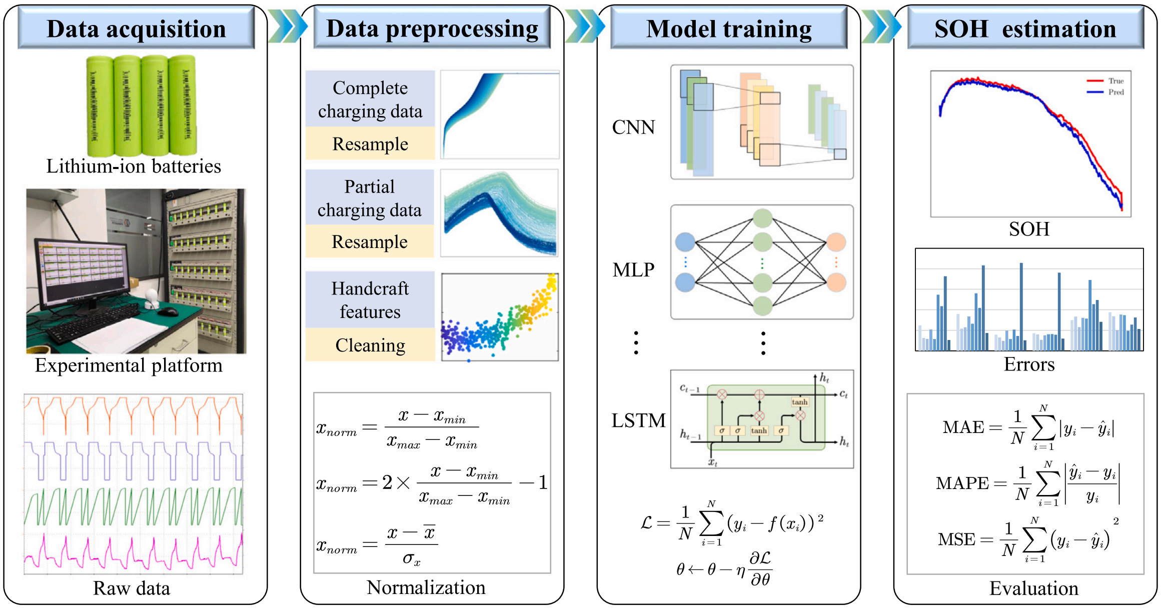

Data acquisition
Lithium-ion batteries
Experimental platform
Raw data
Data preprocessing
Complete charging data
Resample
Partial charging data
Resample
Handcraft features
Cleaning
$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}$
$x_{norm} = 2 \times \frac{x - x_{min}}{x_{max} - x_{min}} - 1$
$x_{norm} = \frac{x - \overline{x}}{\sigma_x}$
Normalization
Model training
CNN
MLP
LSTM
$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}(y_i - f(x_i))^2$
$\theta \leftarrow \theta - \eta\frac{\partial\mathcal{L}}{\partial\theta}$
SOH estimation
True
Pred
SOH
Errors
$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$
$\mathrm{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{\hat{y}_i - y_i}{y_i}\right|$
$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$
Evaluation

Fig. 8. The flow chart of battery SOH estimation. .

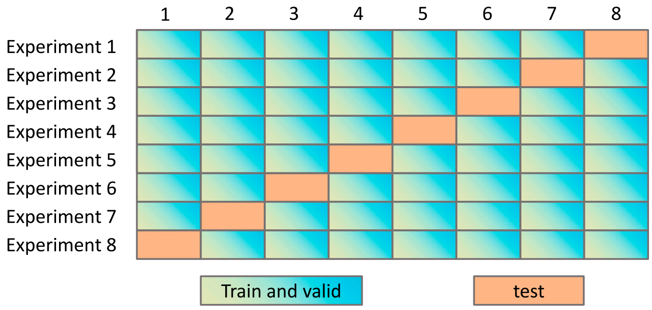


Fig. 9. Example visualization of data splitting strategy for batch 1 in XJTU battery dataset. Batch 1 consists of 8 batteries. Each battery was used in turn as a test battery to carry out experiments.

consisting of 45 batteries per set. A total of 18 sets of preliminary experimental results were generated. However, due to space limitations, we only presented tables for two sets of preliminary experimental results, one for each dataset. The remaining results have been processed and condensed into fewer tables.

To be specific, the preliminary experimental results of the two datasets are given in Tables 4 and 5. The inputs for the results in both tables are handcraft features with [−1, 1] normalization. The other preliminary experimental results will be placed in our code repository, which are available from this link.

7.1. Detailed results of the XJTU battery dataset

The results of the XJTU battery dataset are shown in Table 4, Table D.8 to Table D.10 in Appendix D. Table 4 details the test results for the 55 batteries, where each model takes handcraft features with [−1, 1] normalization as input. The results from Table D.8 to Table D.10 are the results of batch average. Specifically, the MAE, MAPE, and MSE of each batch in these tables are the average of all batteries in this batch. Notably, to avoid accidents, all results in all tables are averaged from three experiments.

It can be seen from Table 4 that the SOH estimation difficulty of each batch varies. Under the premise of [−1,1] normalized handcraft features as input, the estimation difficulty of batch 1, batch 3, and batch 4 is relatively low, since the estimation errors of the batteries in these batches are basically in the same order of magnitude and smaller. Most of the batteries in batch 2 also have small errors, except for batteries 2, 3, 8, 9, 14, and 15. It can be seen that the estimation errors of these batteries on all models are significantly larger than those of other batteries in batch 2. The SOH estimation of batch 5 and batch 6 is more difficult than other batches, because the errors of these batteries are generally larger than those of other batches. The fact is that the charging and discharging strategies of batch 5 and batch 6 are more complicated. From the model perspective, MLP, GRU, and CNN all perform well, while LSTM and Attention are poor.

Table D.8 shows the results of the XJTU battery dataset with [−1, 1] normalization. From the perspective of input type, the best results are achieved in most experiments with handcraft features as input. From the perspective of model, MLP gets the best results, followed by CNN. It is worth noting that in all experiments with complete charging data as input, the errors of LSTM and GRU are significantly larger than those of other models. In experiments with the other two input types, this gap was less pronounced.

Table D.9 shows the results of the XJTU battery dataset with [0, 1] normalization. Overall, the best results are still obtained in experiments with handcraft features as input. From the perspective of model, CNN obtains the best results, followed by MLP and GRU. Similar results to Table D.8 can be obtained in that the errors of LSTM and GRU with complete charging data as input are obviously larger than those of other models. The possible fact is that the difference between [−1, 1] normalization and [0, 1] normalization is small. Another noteworthy point is that in all experiments with CNN model, the experiments with partial charging data as input get the best results.

Table D.10 shows the results of the XJTU battery dataset with z-score normalization. Contrary to the results of the previous two tables, CNN performs the best, followed by Attention, and MLP is the worst of all models. From the perspective of input type, most of the experiments with handicraft features as input get the best results, except that the best results of MLP appear in the experiments with complete charging data as input. In addition, by comparing Table D.8, Table D.9, and Table D.10, the results of z-score normalization seem to be worse than those of the other two normalization methods.

Table 4
The results of XJTU battery dataset, the input type is handcraft features with [−1, 1] normalization. The best result was **bolded**.

Model		CNN				LSTM				GRU				MLP				Attention			
Batch	Battery	MAE	MAPE	MSE	R2	MAE	MAPE	MSE	R2	MAE	MAPE	MSE	R2	MAE	MAPE	MSE	R2	MAE	MAPE	MSE	R2
1	1	**5.162**	**5.703**	0.057	0.976	5.901	6.485	**0.054**	**0.977**	5.716	6.354	0.055	**0.977**	5.818	6.480	0.058	0.975	9.193	10.145	0.133	0.944
	2	5.383	5.964	0.047	0.983	4.960	5.559	0.045	0.984	3.612	4.034	0.024	0.991	**3.405**	**3.811**	**0.021**	**0.993**	5.959	6.587	0.070	0.975
	3	5.594	6.245	0.068	0.977	4.971	5.556	0.060	0.979	3.988	4.469	0.044	0.985	**2.940**	**3.297**	**0.039**	**0.987**	8.229	8.974	0.133	0.954
	4	3.568	3.942	0.025	0.991	4.079	4.461	0.024	0.991	**3.103**	**3.318**	**0.013**	**0.995**	3.947	4.150	0.022	0.992	5.904	6.443	0.058	0.979
	5	5.468	5.919	0.046	0.985	3.637	4.046	0.024	0.992	2.507	2.753	0.010	0.997	**2.126**	**2.307**	**0.007**	**0.998**	6.384	7.058	0.072	0.976
	6	6.015	6.506	0.051	0.982	3.172	3.477	0.018	0.994	**2.666**	**2.888**	**0.011**	**0.996**	4.695	4.968	0.028	0.990	4.922	5.291	0.036	0.987
	7	**6.000**	**6.357**	**0.050**	**0.983**	6.703	7.198	0.062	0.979	6.397	6.822	0.053	0.982	7.761	8.288	0.072	0.976	6.288	6.913	0.068	0.977
	8	9.335	9.854	0.139	0.938	3.221	3.530	0.022	0.990	**2.910**	**3.149**	**0.016**	**0.993**	3.341	3.592	0.022	0.990	5.966	6.494	0.056	0.975
2	1	**2.686**	**3.002**	0.020	0.990	4.903	5.262	0.035	0.982	3.063	3.282	**0.017**	**0.991**	3.148	3.373	0.020	0.990	5.983	6.483	0.059	0.970
	2	17.913	18.738	0.514	0.959	**12.825**	**13.475**	0.212	0.933	12.880	13.549	**0.191**	**0.968**	13.775	14.503	0.207	0.964	12.929	13.905	0.270	0.920
	3	23.021	24.294	0.864	0.951	19.301	20.348	0.454	0.905	18.954	19.978	0.445	0.942	20.828	21.960	0.544	**0.981**	**15.894**	**16.891**	**0.338**	0.877
	4	6.893	7.394	0.082	**0.977**	6.360	6.861	0.064	0.951	5.259	5.678	0.046	0.959	**3.802**	**4.113**	**0.028**	0.976	11.143	12.175	0.214	0.913
	5	22.241	23.427	0.929	0.771	5.248	5.523	0.040	0.978	**3.610**	**3.819**	**0.024**	**0.983**	3.714	3.948	0.026	0.982	7.282	7.946	0.093	0.932
	6	8.498	9.161	0.138	0.859	6.556	7.063	0.065	0.881	4.204	4.525	0.030	0.924	**3.585**	**3.833**	**0.025**	**0.965**	6.323	6.773	0.069	0.904
	7	6.985	7.556	0.077	0.767	6.099	6.411	0.051	0.750	**5.122**	**5.496**	**0.038**	0.818	5.548	5.992	0.049	**0.875**	7.187	7.926	0.103	0.685
	8	17.726	18.712	0.508	0.736	10.147	10.809	0.135	0.891	10.025	10.735	0.123	**0.902**	9.340	10.033	0.108	0.894	**6.125**	**6.603**	**0.060**	0.861
	9	11.339	12.240	0.186	0.662	**8.385**	**9.241**	**0.089**	0.823	10.734	11.627	0.130	0.826	12.087	12.961	0.177	0.787	13.890	15.059	0.226	**0.868**
	10	6.001	6.602	0.066	0.961	7.007	7.969	0.106	0.970	**4.628**	**5.235**	**0.051**	0.978	6.342	6.917	0.058	**0.987**	8.487	9.520	0.127	0.899
	11	6.083	6.630	0.069	0.575	10.851	11.906	0.135	0.981	8.169	8.925	0.082	**0.989**	**3.796**	**4.152**	**0.028**	0.988	12.401	13.609	0.174	0.958
	12	4.960	5.369	**0.033**	0.925	7.390	8.053	0.070	0.965	6.494	7.013	0.059	0.984	**4.640**	**5.018**	0.034	**0.987**	9.944	10.701	0.124	0.963
	13	14.022	15.093	0.375	0.972	4.148	4.609	0.036	0.982	**3.300**	**3.647**	**0.028**	**0.986**	3.651	4.025	0.029	0.982	7.806	8.631	0.112	0.963
	14	10.306	11.180	0.203	0.661	12.425	13.672	0.171	0.910	9.836	10.801	0.109	0.918	**5.823**	**6.345**	**0.051**	0.928	11.011	12.098	0.138	**0.960**
	15	13.320	15.005	0.342	0.891	18.403	20.382	0.367	**0.948**	15.956	17.553	0.268	0.924	**12.990**	**14.212**	**0.183**	0.896	20.712	22.890	0.462	0.867
3	1	4.407	4.730	0.035	0.989	4.897	5.368	0.042	0.987	3.460	3.746	0.021	0.994	**2.965**	**3.191**	**0.016**	**0.995**	4.966	5.391	0.045	0.986
	2	3.760	4.112	0.024	0.993	4.321	4.736	0.034	0.990	2.933	3.224	0.018	**0.995**	**2.925**	**3.203**	**0.015**	**0.995**	4.967	5.478	0.048	0.986
	3	**3.279**	**3.495**	**0.017**	**0.993**	5.507	5.828	0.040	0.984	5.434	5.784	0.037	0.985	4.315	4.646	0.027	0.989	9.348	9.951	0.112	0.954
	4	10.425	10.827	0.146	0.953	9.088	9.484	0.105	0.966	7.508	7.787	0.073	0.976	7.292	7.548	0.069	0.978	**6.132**	**6.551**	**0.061**	**0.980**
	5	**3.191**	**3.414**	**0.016**	**0.995**	4.623	5.016	0.034	0.989	3.872	4.155	0.022	0.993	3.994	4.263	0.023	0.993	4.456	4.768	0.033	0.990
	6	4.350	4.652	0.025	0.992	3.317	3.537	0.018	0.994	**2.862**	**3.015**	**0.012**	**0.996**	3.644	3.833	0.018	0.994	5.654	5.965	0.046	0.984
	7	3.934	4.212	0.028	0.989	3.544	3.812	0.027	0.989	**3.001**	**3.229**	0.018	**0.993**	3.043	3.256	**0.016**	**0.993**	6.276	6.713	0.078	0.969
	8	4.920	5.206	0.047	0.981	2.511	2.663	**0.010**	**0.996**	**2.436**	**2.614**	0.010	**0.996**	2.469	2.655	0.011	**0.996**	5.512	5.837	0.044	0.982
4	1	7.206	7.792	0.062	0.973	**4.502**	**5.005**	0.043	0.981	4.672	5.108	**0.036**	**0.985**	5.236	5.653	0.041	0.982	7.105	7.687	0.080	0.966
	2	**4.213**	**4.648**	0.047	0.977	8.359	8.996	0.092	0.955	7.164	7.690	0.066	0.968	4.461	4.812	**0.033**	**0.984**	5.508	5.991	0.060	0.971
	3	**3.810**	**4.086**	**0.023**	**0.991**	7.180	7.745	0.072	0.972	6.027	6.440	0.051	0.981	5.510	5.835	0.044	0.983	5.559	5.962	0.049	0.981
	4	6.025	6.396	0.043	0.982	5.912	6.359	0.051	0.978	4.874	5.231	0.036	0.985	**4.519**	**4.816**	**0.031**	**0.987**	5.458	5.951	0.055	0.977
	5	**3.793**	**4.090**	**0.021**	**0.990**	5.213	5.622	0.044	0.978	5.132	5.532	0.040	0.980	4.486	4.854	0.031	0.984	6.974	7.550	0.074	0.963
	6	4.876	5.262	**0.034**	**0.980**	4.424	4.800	0.037	0.979	**4.202**	**4.579**	0.035	**0.980**	4.484	4.896	0.038	0.978	4.578	4.930	0.039	0.978
	7	8.820	9.506	0.098	0.947	7.413	8.035	0.074	0.960	7.407	7.982	0.070	0.962	**7.143**	**7.669**	**0.064**	**0.965**	8.962	9.499	0.111	0.940
	8	4.060	4.463	0.030	0.986	4.627	5.084	0.043	0.980	**3.715**	4.088	0.027	0.987	3.721	**4.056**	**0.025**	**0.989**	5.973	6.460	0.056	0.974
5	1	16.638	18.192	0.482	0.720	**9.297**	**10.185**	0.148	0.914	10.530	11.559	0.162	0.906	23.123	25.187	1.474	0.440	9.343	10.221	**0.144**	**0.916**
	2	11.560	12.360	0.184	0.889	8.563	9.227	0.125	0.925	**6.891**	**7.411**	**0.089**	**0.946**	22.565	24.471	0.610	0.630	7.829	8.458	0.121	0.927
	3	11.237	12.483	0.202	0.908	10.431	11.744	0.208	0.905	**8.776**	**9.868**	**0.151**	**0.931**	11.442	12.510	0.199	0.909	9.535	10.605	0.160	0.927
	4	8.970	9.704	0.141	0.937	6.626	7.290	0.114	0.949	**6.440**	**7.085**	**0.106**	**0.952**	16.997	18.785	0.408	0.817	9.267	10.136	0.171	0.923
	5	13.762	15.480	0.309	0.866	**11.979**	13.386	0.238	0.897	12.868	14.241	0.237	0.897	12.019	**12.926**	**0.209**	**0.909**	14.367	15.746	0.283	0.877
	6	8.009	8.717	0.109	0.946	6.493	7.188	0.095	0.953	**5.963**	**6.586**	**0.079**	**0.961**	8.245	9.009	0.115	0.943	8.795	9.718	0.161	0.921
	7	**7.490**	**8.232**	**0.130**	**0.927**	10.722	11.599	0.157	0.913	11.914	12.884	0.177	0.901	17.775	19.149	0.435	0.757	10.250	11.194	0.153	0.915
	8	14.758	15.973	0.342	0.878	13.020	14.685	0.292	0.896	11.894	13.420	0.229	0.918	**10.148**	**11.095**	**0.155**	**0.945**	12.413	13.875	0.245	0.913
6	1	**5.441**	**5.905**	**0.062**	**0.973**	13.106	14.311	0.281	0.878	11.452	12.516	0.226	0.902	10.761	11.766	0.206	0.911	8.263	9.091	0.149	0.935
	2	**8.820**	**9.496**	**0.130**	**0.924**	10.065	10.862	0.185	0.892	10.750	11.537	0.190	0.888	11.018	11.800	0.193	0.887	11.648	12.474	0.211	0.876
	3	**8.417**	**9.058**	**0.101**	**0.951**	9.571	10.478	0.184	0.911	8.943	9.764	0.157	0.924	8.647	9.436	0.145	0.930	8.974	9.702	0.129	0.938
	4	**11.940**	**12.787**	**0.178**	**0.918**	12.826	14.018	0.275	0.873	12.806	13.893	0.245	0.887	13.396	14.477	0.252	0.883	13.707	14.682	0.240	0.889
	5	21.862	23.974	0.566	0.674	17.011	18.599	0.380	0.782	**15.664**	**17.088**	**0.329**	**0.811**	16.253	17.726	0.353	0.797	21.889	23.853	0.592	0.660
	6	12.039	12.693	**0.204**	**0.917**	13.015	14.166	0.313	0.873	11.155	12.214	0.272	0.889	10.967	11.986	0.263	0.893	**10.773**	**11.742**	0.249	0.899
	7	**5.754**	**6.214**	**0.061**	**0.962**	9.466	10.334	0.170	0.893	8.540	9.299	0.140	0.913	8.527	9.247	0.137	0.914	8.290	8.996	0.135	0.915
	8	**6.113**	**6.602**	**0.067**	**0.955**	10.877	11.754	0.197	0.870	10.773	11.621	0.183	0.879	10.330	11.131	0.166	0.890	7.675	8.313	0.111	0.927

Note: (1). All values are the average value of 3 experiments; (2). For intuitive display, the values of MAE, MAPE, and MSE are all magnified 1000 times.

7.2. Detailed results of the Toyota-MIT-Stanford dataset

The results of the Toyota-MIT-Stanford dataset are shown in Table 5, Table D.11 to Table D.13 in Appendix D. Table 5 details the test results for the 45 batteries, where each model takes handcraft features with [−1, 1] normalization as input. The results from Table D.11 to Table D.13 are the results of batch average. Specifically, the MAE, MAPE, and MSE of each batch in these tables are the average of 5 batteries in this batch.

The Toyota-MIT-Stanford dataset contains 45 batteries, which are evenly divided into 9 groups. The charging process of each group goes through 4 stages from 0 to 80% SOC, and all batteries are discharged at a constant current of 4 C. In other words, the charging and discharging strategy of Toyota-MIT-Stanford is relatively simple. This also corresponds to the conclusion in Table 5, that is, the SOH estimation results of the 9 groups in the Toyota-MIT-Stanford dataset are relatively uniform, except for batteries 1 and 2 in group 3, and batteries 1, 2, and 4 in group 4. From the perspective of the model, the Attention model has won the first place the most times, and LSTM has the least.

The results of the Toyota-MIT-Stanford dataset with [−1, 1] normalization are shown in Table D.13. It can be seen from the table that the experiment with partial charging data or handcrafted features as input has obtained better results, tied for first place, without distinguishing the model. For CNN model, the results with handcraft features as input are the best, while the MLP model has the best result with partial

Table 5
The results of Toyota-MIT-Stanford dataset, the input type is handcraft features with $[-1, 1]$ normalization. The best result was **bolded**.

Model		CNN				LSTM				GRU				MLP				Attention			
Batch	Battery	MAE	MAPE	MSE	R2	MAE	MAPE	MSE	R2	MAE	MAPE	MSE	R2	MAE	MAPE	MSE	R2	MAE	MAPE	MSE	R2
1	1	7.106	7.723	0.081	0.899	6.797	7.465	0.066	0.917	**4.764**	**5.205**	**0.035**	**0.956**	5.396	5.885	0.044	0.945	10.011	10.970	0.138	0.828
	2	6.197	6.711	0.052	0.959	5.170	5.633	0.041	0.968	4.793	5.217	0.031	0.976	5.695	6.148	0.039	0.970	**3.089**	**3.424**	**0.019**	**0.985**
	3	**4.460**	**4.869**	**0.031**	**0.966**	7.718	8.324	0.075	0.919	9.035	9.711	0.095	0.896	9.278	9.957	0.101	0.890	11.117	12.005	0.153	0.833
	4	8.892	9.537	0.106	0.908	10.051	10.702	0.126	0.892	9.325	9.901	0.106	0.909	10.007	10.615	0.121	0.896	**7.454**	**8.038**	**0.084**	**0.928**
	5	**2.084**	**2.301**	**0.009**	**0.991**	3.585	3.940	0.024	0.976	2.628	2.888	0.013	0.987	3.833	4.172	0.025	0.976	4.897	5.350	0.039	0.962
2	1	10.542	11.137	0.128	0.902	10.328	10.855	0.127	0.902	9.902	10.513	0.110	0.916	11.648	12.379	0.148	0.887	**6.121**	**6.457**	**0.063**	**0.952**
	2	8.628	9.324	0.101	0.907	11.304	12.088	0.167	0.846	9.845	10.532	0.124	0.885	**5.990**	**6.551**	**0.057**	**0.947**	16.205	17.295	0.334	0.692
	3	**3.707**	**4.047**	**0.023**	**0.983**	7.108	7.853	0.095	0.931	5.966	6.538	0.063	0.955	5.486	5.953	0.046	0.967	6.663	7.345	0.082	0.941
	4	5.772	6.097	0.080	0.922	5.670	5.996	0.047	0.954	5.388	5.686	0.046	0.955	**4.629**	**4.913**	**0.038**	**0.963**	5.269	5.695	0.048	0.953
	5	5.633	6.085	0.079	0.921	5.490	6.040	0.052	0.948	6.032	6.593	0.053	0.946	**5.116**	**5.606**	**0.044**	**0.956**	8.369	9.049	0.090	0.909
3	1	**13.441**	**14.599**	**0.231**	**0.817**	19.552	21.576	0.429	0.660	20.368	22.381	0.440	0.652	21.016	22.966	0.459	0.637	20.679	22.687	0.502	0.602
	2	10.517	11.062	0.135	0.877	11.875	12.522	0.159	0.855	13.743	14.541	0.205	0.813	17.850	18.907	0.337	0.693	**8.177**	**8.643**	**0.086**	**0.922**
	3	10.299	11.013	0.142	0.862	8.363	8.851	0.084	0.918	7.599	8.042	0.068	0.934	5.195	5.522	**0.036**	**0.965**	**4.794**	**5.135**	0.036	**0.965**
	4	5.857	6.345	0.060	0.942	6.183	6.583	0.050	0.951	6.566	6.968	0.057	0.944	8.746	9.267	0.095	0.908	**3.967**	**4.269**	**0.025**	**0.976**
	5	9.181	9.792	0.129	0.864	**5.901**	**6.373**	0.050	0.947	6.096	6.570	**0.048**	**0.949**	17.748	18.916	0.372	0.607	6.781	7.343	0.068	0.928
4	1	31.859	33.738	1.631	0.363	15.003	16.140	0.390	0.674	**11.408**	**12.241**	**0.164**	**0.863**	60.077	63.804	4.664	0.596	13.605	14.567	0.279	0.767
	2	12.158	13.217	0.214	0.706	12.641	13.784	0.189	0.740	12.550	13.620	0.180	0.753	**10.529**	**11.382**	**0.138**	**0.810**	13.881	15.055	0.237	0.675
	3	**8.506**	**9.251**	**0.108**	**0.863**	16.641	18.045	0.296	0.622	15.810	17.070	0.266	0.660	9.687	10.470	0.140	0.821	19.946	21.440	0.441	0.437
	4	19.688	20.784	0.445	0.579	15.108	15.772	0.263	0.751	17.563	18.390	0.339	0.679	20.519	21.466	0.482	0.544	**10.488**	**11.023**	**0.130**	**0.877**
	5	27.700	29.321	0.843	0.630	6.257	6.609	**0.045**	**0.943**	9.003	9.558	0.089	0.888	30.656	32.340	1.019	0.585	**4.654**	**5.066**	0.049	0.939
5	1	7.870	8.576	0.111	0.905	6.347	7.076	0.087	0.925	**6.183**	**6.789**	**0.063**	**0.946**	9.786	10.572	0.110	0.905	8.503	9.309	0.116	0.901
	2	8.152	8.988	0.122	0.863	7.869	8.651	0.093	0.896	**5.677**	**6.220**	**0.050**	**0.944**	6.137	6.743	0.059	0.934	14.391	15.595	0.257	0.713
	3	**8.552**	**9.081**	**0.092**	**0.924**	11.892	12.600	0.168	0.862	12.243	12.983	0.170	0.861	12.855	13.663	0.185	0.848	12.601	13.544	0.225	0.816
	4	9.968	10.628	0.217	0.826	10.382	10.922	0.129	0.896	7.029	7.405	0.063	**0.949**	16.596	17.506	0.295	0.763	**6.515**	**6.961**	**0.063**	**0.949**
	5	6.946	7.558	0.100	0.895	7.394	7.948	0.071	0.926	6.726	7.223	0.057	0.940	**3.322**	**3.594**	**0.019**	**0.980**	10.295	11.064	0.141	0.852
6	1	8.067	8.509	0.082	0.919	**2.676**	**2.900**	0.014	0.986	2.821	3.006	**0.012**	**0.988**	3.059	3.261	0.014	0.987	5.865	6.300	0.050	0.950
	2	10.831	11.426	0.166	0.851	8.165	8.585	0.081	0.928	5.860	6.171	0.044	0.960	6.569	6.951	0.057	0.949	**4.375**	**4.685**	**0.038**	**0.966**
	3	**8.068**	**8.755**	**0.088**	**0.904**	11.410	12.212	0.170	0.814	12.104	12.911	0.184	0.798	12.253	13.093	0.177	0.807	14.525	15.611	0.251	0.726
	4	8.330	8.819	0.094	0.907	9.622	10.119	0.110	0.891	8.937	9.365	0.095	0.907	9.687	10.148	0.110	0.892	**5.654**	**6.010**	**0.051**	**0.950**
	5	8.247	8.871	0.096	0.892	3.225	3.580	0.027	0.970	**2.806**	**3.070**	**0.015**	**0.983**	3.570	3.869	0.021	0.977	6.279	6.787	0.057	0.936
7	1	6.864	7.479	0.075	0.950	4.613	5.124	0.042	0.972	3.905	4.322	0.027	0.982	**3.122**	**3.466**	**0.020**	**0.987**	9.671	10.646	0.127	0.915
	2	12.822	13.866	0.237	0.805	9.915	10.578	0.121	0.901	11.559	12.374	0.151	0.876	34.755	37.476	1.347	0.705	**7.234**	**7.781**	**0.072**	**0.941**
	3	9.830	10.658	0.152	0.845	3.459	3.742	0.016	0.983	**2.884**	**3.135**	**0.013**	**0.987**	4.710	5.183	0.039	0.960	4.134	4.468	0.027	0.973
	4	3.748	4.155	0.026	0.978	4.341	4.783	0.037	0.969	**3.123**	**3.450**	**0.018**	**0.985**	4.606	5.081	0.031	0.975	12.716	13.835	0.194	0.838
	5	7.486	8.191	0.098	0.934	5.150	5.755	0.063	0.957	**4.219**	**4.694**	**0.038**	**0.974**	6.733	7.271	0.061	0.959	5.870	6.541	0.075	0.949
8	1	14.929	15.965	0.259	0.755	13.224	14.032	0.204	0.807	12.706	13.449	0.184	0.826	11.698	12.391	0.158	0.851	**7.168**	**7.658**	**0.074**	**0.930**
	2	7.683	8.290	0.086	0.902	10.973	11.911	0.144	0.837	9.813	10.579	0.111	0.874	**6.562**	**7.072**	**0.053**	**0.940**	16.346	17.609	0.302	0.658
	3	6.718	7.268	0.067	0.934	5.096	5.553	0.041	0.959	3.925	4.259	0.023	0.978	**3.815**	**4.126**	**0.019**	**0.982**	8.525	9.246	0.098	0.904
	4	17.040	18.058	0.349	0.636	16.142	17.045	0.292	0.696	18.459	19.561	0.364	0.621	20.035	21.244	0.429	0.553	**7.372**	**7.830**	**0.071**	**0.927**
	5	15.972	17.274	0.333	0.644	14.569	15.861	0.253	0.729	17.727	19.204	0.342	0.634	17.004	18.356	0.326	0.652	**12.553**	**13.672**	**0.228**	**0.756**
9	1	6.594	7.203	0.092	0.858	**5.309**	**5.727**	**0.037**	**0.942**	6.569	7.098	0.050	0.922	6.324	6.942	0.102	0.841	6.786	7.429	0.082	0.872
	2	10.876	11.558	0.142	0.815	11.733	12.434	0.157	0.797	13.484	14.278	0.205	0.734	18.935	20.055	0.399	0.483	**5.140**	**5.557**	**0.058**	**0.924**
	3	9.402	10.140	0.162	0.809	5.405	5.847	0.049	0.943	**4.471**	**4.823**	**0.033**	**0.962**	38.017	40.387	1.616	0.698	5.871	6.365	0.056	0.934
	4	**4.485**	**4.917**	**0.041**	**0.946**	6.249	6.872	0.060	0.921	7.420	8.052	0.063	0.917	6.528	7.092	0.060	0.921	8.834	9.573	0.093	0.878
	5	15.319	16.273	0.280	0.683	7.967	8.612	0.093	0.894	**6.255**	**6.776**	0.060	0.931	30.935	32.724	1.051	0.592	6.283	6.801	**0.060**	**0.932**

Note: (1). All values are the average value of 3 experiments; (2). For intuitive display, the values of MAE, MAPE, and MSE are all magnified 1000 times.

charging data as input. For the other three models, similar results are achieved with handcraft features as input and partial charging data as input. Counting the results of all models in all input types, it can be found that the Attention model has achieved the most first places, while LSTM has the least.

Table D.12 shows the results of the Toyota-MIT-Stanford dataset with [0, 1] normalization. From the perspective of input type, the best results are obtained in most experiments with partial charging data as input, followed by handcraft features, and last by complete charging data. However, overall, the gap in the experimental results among the three input types is not large. For CNN model, the results with complete charging data as input are the best. From a model perspective, Attention model still wins the most first places, while CNN has the least.

The results of the Toyota-MIT-Stanford dataset with z-score normalization are given in Table D.13. Overall, the best results are obtained in experiments with handcraft features as input. For each model, complete charging data is the best input type for MLP, while partial charging data is the best for GRU. From a model perspective, contrary to the previous two tables, CNN and MLP perform the best, and LSTM is the worst.

7.3. Results of models

From the results, we can observe that CNN model is a competitive model on the XJTU battery dataset. MLP model gets promising results on the XJTU battery dataset when the normalization method is [−1, 1] normalization or [0, 1] normalization. However, it obtains the worst results when the z-score normalization method is used to preprocess data.

Attention model achieves great success on the Toyota-MIT-Stanford dataset when the normalization method is [−1, 1] normalization or [0, 1] normalization, while CNN and MLP perform better when the data is preprocessed with z-score normalization.

7.4. Results of input types

In two datasets and all normalization methods, handcraft features seem to be a promising input type. For the XJTU battery dataset, the charging and discharging strategy is complex. Therefore, the handcraft features are more suitable for most situations, since the complete charging data or partial charging data may contain more noise, while handcrafted features are simply cleaned. For the Toyota-MIT-Stanford dataset, the charging and discharging strategy is simple, so the gap among three input types is not large. In Tables D.11 and D.12, experiments with partial charging data as input achieve even better results than those with handcraft features as input.

7.5. Results of normalization method

To provide a comprehensive and intuitive representation, we have plotted the test MAE for all models with three input types and three normalization methods, as shown in Fig. 10. It can be seen from figure that [−1, 1] normalization and [0, 1] normalization are better than z-score normalization on almost all models, regardless of the input type. In most cases, [−1, 1] normalization is slightly better than [0, 1] normalization. Therefore, in future research, it is recommended to use [−1, 1] normalization method for data preprocessing.

8. Discussion

8.1. Discussion on dataset

Although there are many publicly available datasets, most of them adopt the strategy of constant current and constant voltage charging mode, and constant current discharging mode. In addition, most publicly available datasets are fully charged and fully discharged. In actual usage, users often do not fully charge or discharge. Therefore, it is necessary to publish more datasets in line with actual usage scenarios for scholars to study. In this paper, we published a batch of dataset that simulate actual working scenarios of GEO satellites, as well as a batch of dataset that simulate the actual usage of users, that is, the battery discharges at random currents for random durations. More datasets that are more in line with actual usage scenarios are needed, such as completely random current for charging and discharging with not fully charged and not fully discharged.

8.2. Discussion on features

In this paper, we compare the impact of three input types on the results: complete charging data, partial charging data, and handcraft features. In each input type there are points worth studying. For example, in the study that takes partial charging data as input, how to determine the starting and ending point of input data. In this paper, we subjectively truncated the data at the voltage of 3.7–4.1 V. In addition to this truncation method, countless truncation methods can also be designed. How to truncate to maximize the effective information retained and ensure the best performance of the model is a problem worth studying. In the study with handcraft features as input, we extracted 67 statistical features from charging data. Likewise, we firmly believe that better features can be designed. How to extract more reasonable, representative, and universal features is also a direction worth studying.

8.3. Future works

Although DL-based method can obtain promising results in SOH estimation of lithium-ion battery, there are still some issues that merit further discussion and research. In this paper, we further discuss the following four issues, including unsupervised learning, transfer learning, interpretability, and physics-informed machine learning.

8.3.1. Unsupervised learning

Currently, most studies on lithium-ion battery SOH estimation using DL belong to the category of supervised learning methods. They assume that there are sufficient data and labels for model training. In fact, data in real industries are mostly unlabeled. Therefore, how to use a large amount of unlabeled data to train a model is a problem worthy of discussion and research.

Unsupervised learning has achieved great success in computer vision and natural language processing, but little research has been done in the field of battery SOH estimation [134,135]. The DL algorithm is used in various fields due to its ability to extract features from massive amounts of data. However, in the massive data generated in real industries, labeled data is only the tip of the iceberg. Using a large amount of unlabeled data to train the model, DL-based method can exert more powerful capabilities. Unsupervised learning is an alternative approach to this task. How to design a suitable loss function or design a surrogate task to use unlabeled data is a point worth studying in the future.

8.3.2. Transfer learning

The SOH estimation models based on DL follow an assumption that the training set and test set obey the same distribution. However, such an assumption is often not satisfied in reality, as changes in operating conditions will inevitably lead to different sample structures and different data distributions, also called domain shifts or distribution shifts.

The performance of DL models degrades rapidly when test conditions change. Transfer learning is an alternative approach to this problem [136]. In the published work, there are also some articles [137–141] that consider the distribution shift, and propose a SOH estimation model based on transfer learning. How and what to transfer is, however, a question worthy of study. Only by figuring out what the transfer learning model has learned and transferred, can we essentially improve the performance of transfer learning model, thereby avoiding negative transfer.

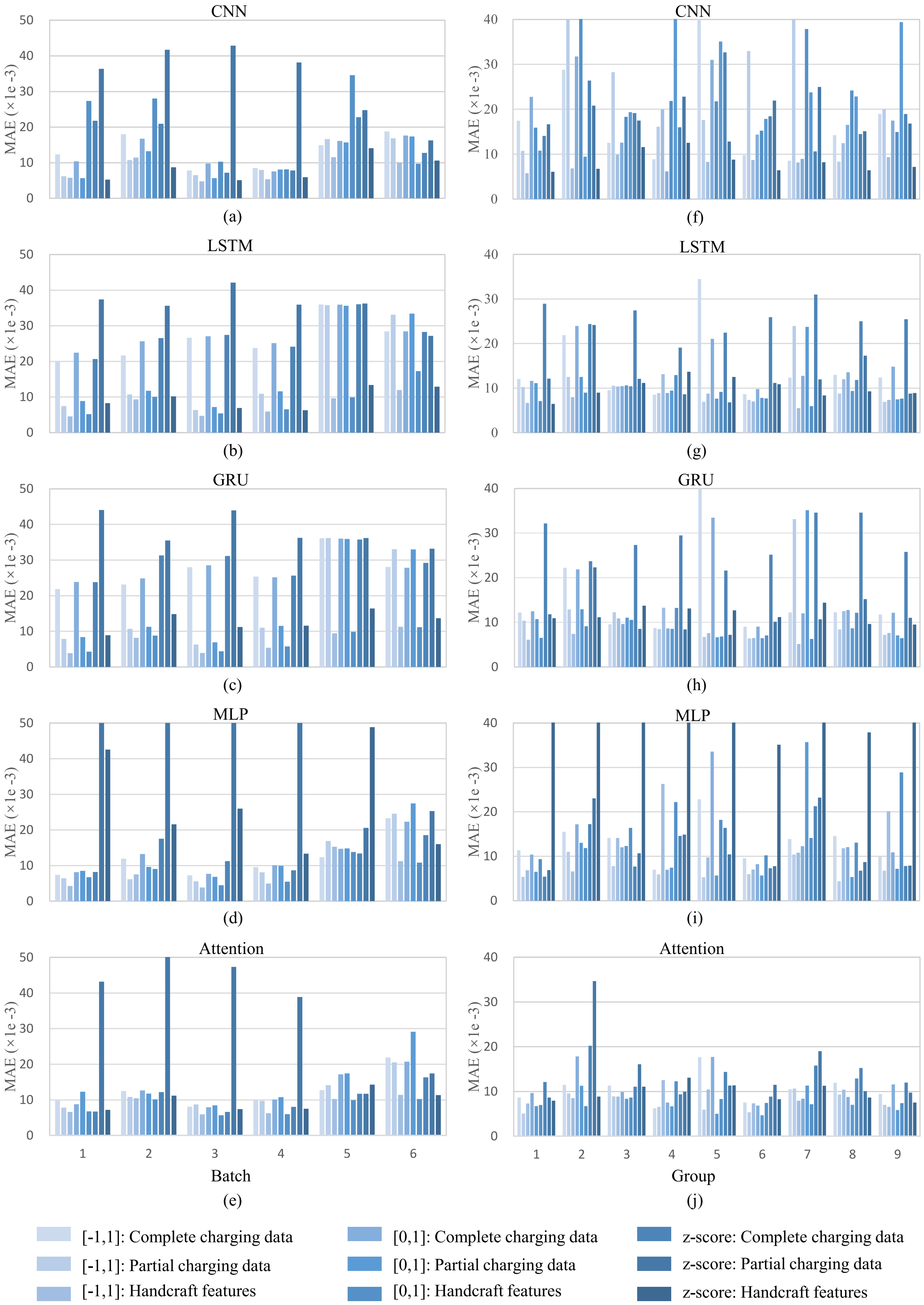
8.3.3. Interpretability

Although DL algorithms can achieve great results, they all lack interpretability and transparency. Due to the “black-box” nature of DL, we do not know whether the model has actually learned the features that can reflect battery degradation, why the model makes such a decision, and when the model will fail, which also limits the deployment of DL models in real industry. Users also tend to be reluctant to use models they cannot explain. In risk-sensitive scenarios, once the model fails or gives wrong prediction results, it will threaten life safety and cause huge property losses. Therefore, model interpretability in SOH estimation is a worthy research issue.

Among the published works, the interpretability of SOH estimation models is less studied. Several papers [142–144] have studied the interpretability of models. In previous work, we proposed an explainability-driven model improvement framework for lithium-ion battery SOH estimation [145]. To the best of our knowledge, this is also the first study of using explanations to improve model performance in SOH estimation tasks. Interpretable models can constitute a human-in-the-loop system in which machine–human interactions can lead to optimal learning. How to explain the decisions of a model and how to use these explanations to improve the model performance are future research trends.

8.3.4. Physics-informed machine learning

Physics-driven machine learning (PIML), or physics-informed neural network (PINN), is a new paradigm developed in recent years. It combines the excellent characteristics of neural networks and the knowledge of physical equations, aiming to more accurately describe and predict physical phenomena and system behavior. On the one hand, PIML generally incorporates physical equations, which makes the model interpretable to a certain extent. On the other hand, DL models



(a) (f)

(b) (g)

(c) (h)

(d) (i)

(e) (j)

Fig. 10. Effect of different normalization methods on model performance. (a)–(e): XJTU dataset. (f)–(j): Toyota-MIT-Stanford dataset. For consistency, the y-axis of all subplots corresponding to the same dataset is constrained to the same value (50 for XJTU battery dataset and 40 for Toyota-MIT-Stanford dataset), and the excess will not be displayed.

can be trained with fewer samples due to the limitation of physical knowledge, and generally, the model can converge faster. Therefore, PINN is a research hotspot in the future.

Among the existing works, some articles [146–149] have initially incorporated physics knowledge in DL models. However, these studies still have some room for improvement. They need to know the parameters of the physical model, or only use the physical knowledge to preprocess the data. Bazant et al. [150] divide models that integrate physical knowledge and DL into five categories: (1) using machine learning to learn the residuals predicted by the physics-based model, (2) using the data generated by the physics-based model to assist in training the machine learning model, (3) using the machine learning model to learn the parameters of the physics-based model, (4) using the physics-based model to constrain the architecture or loss of machine learning models, and (5) using machine learning models to assist in solving physical equations. How to integrate physical knowledge into DL models more subtly and appropriately is worth studying.

9. Conclusion

In this paper, we develop a large lithium-ion battery dataset with run-to-failure data of 55 batteries under 6 charging and discharging strategies. Combined with another publicly available dataset, we evaluate the performance of 5 DL models on 100 batteries. In addition, we also compare the impact of 3 input types and 3 normalization methods on model performance. It is worth noting that the results and conclusions we obtained are only based on the two datasets and feature extraction methods we used, and may not be necessarily representative. Our work aims to evaluate the performance of commonly used DL-based SOH estimation models from different perspectives and provide a benchmark for other scholars to compare their proposed models, avoiding ineffective improvements.

Additionally, we also provide a code library for other scholars to test the performance of their own models and their own dataset. We hope that the evaluation results and code library will facilitate a better understanding of DL-based SOH estimation models and provide a unified framework for proposing more effective models. Further, we discuss future research directions and summarize four directions that may facilitate the development of SOH estimation of lithium-ion battery.

CRediT authorship contribution statement

Fujin Wang: Conceptualization, Methodology, Software, Experiment, Writing – original draft. **Zhi Zhai:** Conceptualization, Methodology, Writing – review & editing. **Bingchen Liu:** Investigation, Writing – original draft. **Shiyu Zheng:** Investigation, Writing – original draft. **Zhibin Zhao:** Conceptualization, Methodology, Writing – review & editing. **Xuefeng Chen:** Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

I have shared the links to my data and code at the manuscript.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grand 52105116 and Grand U22B2013; the China Postdoctoral Science Foundation under Grand 2021M692557 and Grand 2021TQ0263.

Appendix A. Dataset

GEO satellites experience the Earth's shadow during the spring and autumn equinoxes each year, approximately 23 days before and after the equinoxes, resulting in a total duration of approximately 46 days for each occurrence [133]. During the period of the Earth's shadow, the duration of the shadow varies in a regular pattern every day, as depicted in Table A.6 and Fig. A.11. Specifically, the duration of the shadow initially increases gradually before decreasing. An illustration of the discharge capacity curve of battery 1 in batch 6 is given in Fig. A.12.

Appendix B. Feature engineering

A total of 67 features are derived from the charging data. The description of each feature is given in Table B.7. To provide a simple and representative representation, we visualize the features of the battery 1 in the XJTU battery dataset batch 1, as shown in Figs. B.14 and B.15. The specific calculation process of some features will be introduced below.

Cumulate charge energy (E) is calculated by:

$$E = \int_{t_{start}}^{t_{end}} V \cdot I dt. \tag{B.1}$$

where V and I represent voltage and current during the charging process.

Cumulate charge capacity (Q) is calculated by:

$$E = \int_{t_{start}}^{t_{end}} I dt. \tag{B.2}$$

From a statistical point of view, four features are derived, signal mean, standard deviation, skewness and kurtosis. They are calculated based on the following equations:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x(i) \tag{B.3}$$

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} (x(i)-\bar{x})^2} \tag{B.4}$$

$$skewness = \frac{\sum_{i=1}^{n} (x(i)-\bar{x})^3}{(n-1)\sigma^3} \tag{B.5}$$

$$kurtosis = \frac{\sum_{i=1}^{n} (x(i)-\bar{x})^4}{(n-1)\sigma^4} \tag{B.6}$$

The entropy of curve represents the amount of information contained in a curve. Thus, we calculated the entropy of normalized CCCV-CCCT and CVCC-CVCT curves as input features according to the following formula:

$$entropy = -\sum_{i=1}^{n} p_i \cdot \log\left(p_i\right) \tag{B.7}$$

where p_i is normalized value of a curve. The features constant current charge time (CCCT) and constant voltage charge time (CVCT) represent the time span of the two horizontal segments in Fig. B.13. The slopes of CCCV-CCCT and CVCC-CVCT curves are calculated as dV/dt and dI/dt.

Appendix C. Model structure

See Figs. C.16 and C.17.

Appendix D. Results

See Table D.13.

Table A.6
The discharge duration of each cycle of the GEO satellite in the Earth's shadow period [133].

Cycle number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24
Discharge duration	5	20	34	41	46	50	54	56	58	60	62	64	68	69	70	71	72	72	72	72	72	72	72

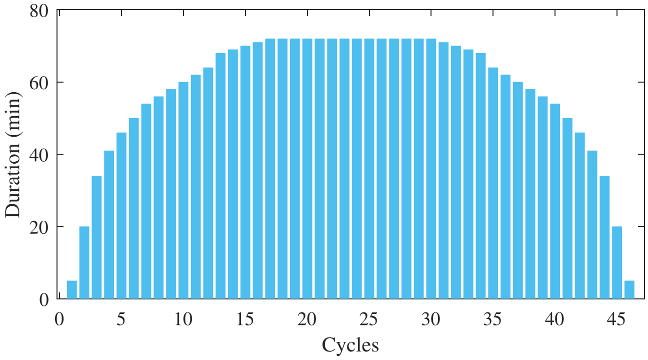


Fig. A.11. The discharge duration of each cycle of the GEO satellite in the Earth's shadow period.

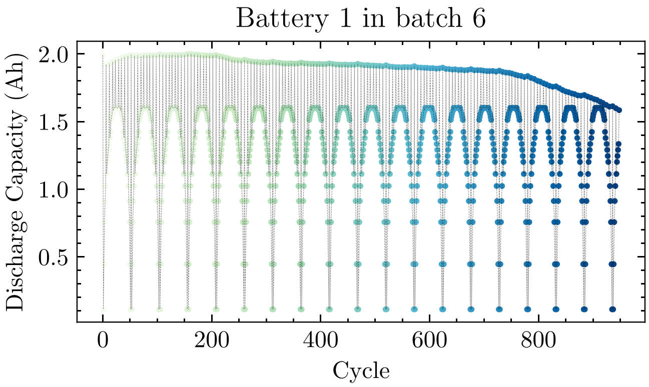


Fig. A.12. An illustration of the discharge capacity curve of battery 1 in batch 6 during the whole life cycles.

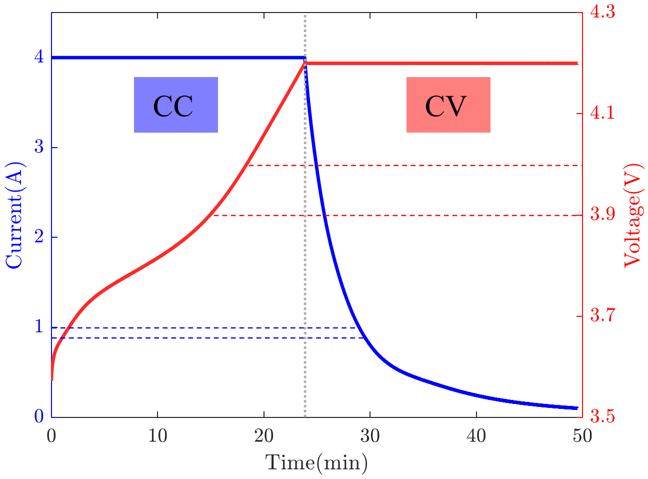


Fig. B.13. Constant current–constant voltage charge strategy. CC represents constant current and CV represents constant voltage.

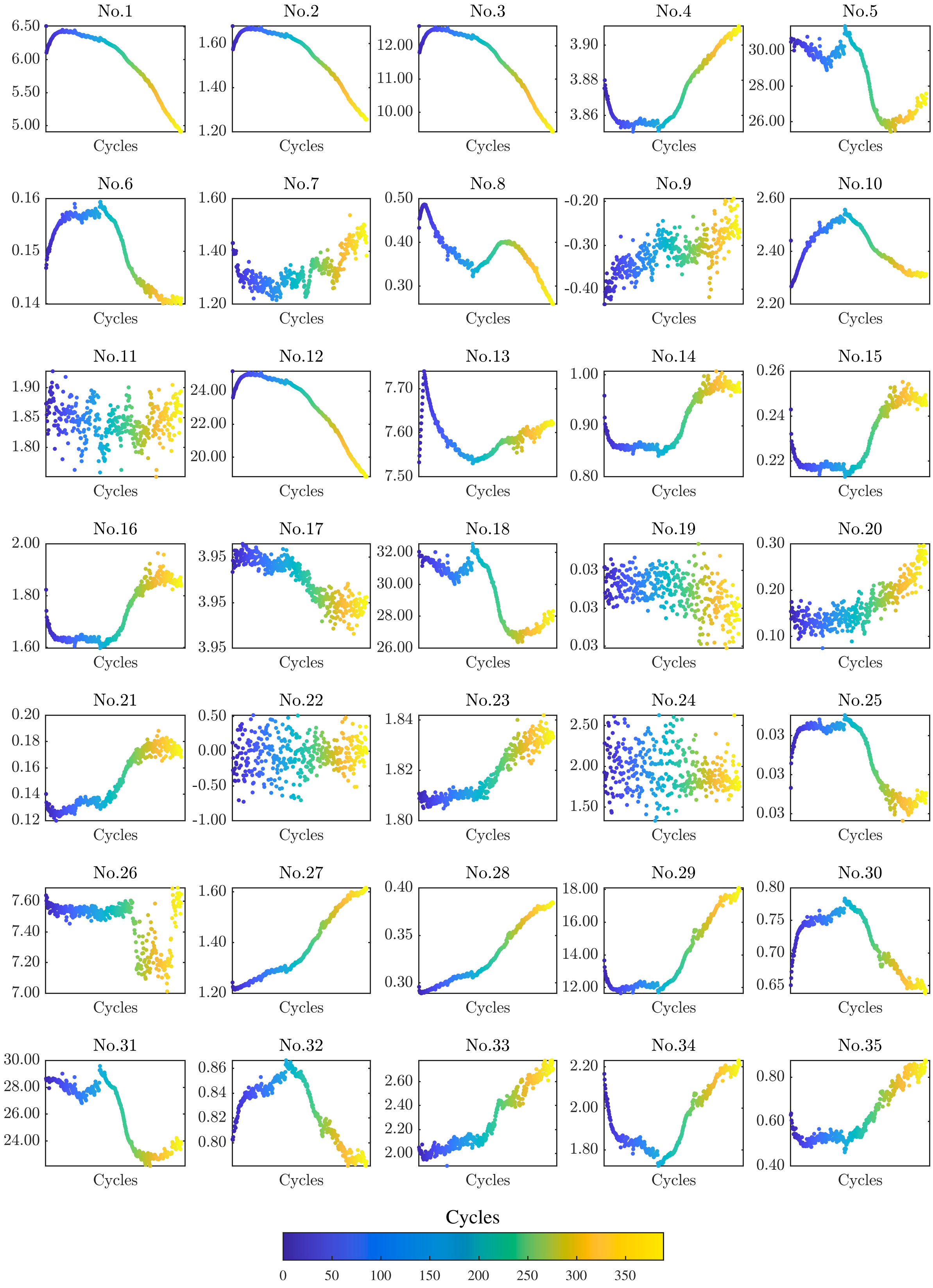


Fig. B.14. A visualization of partial handcraft features for batch 1 battery 1 in XJTU battery dataset. Subheadings No. 1 to No. 35 correspond to Feature No. in Table B.7.

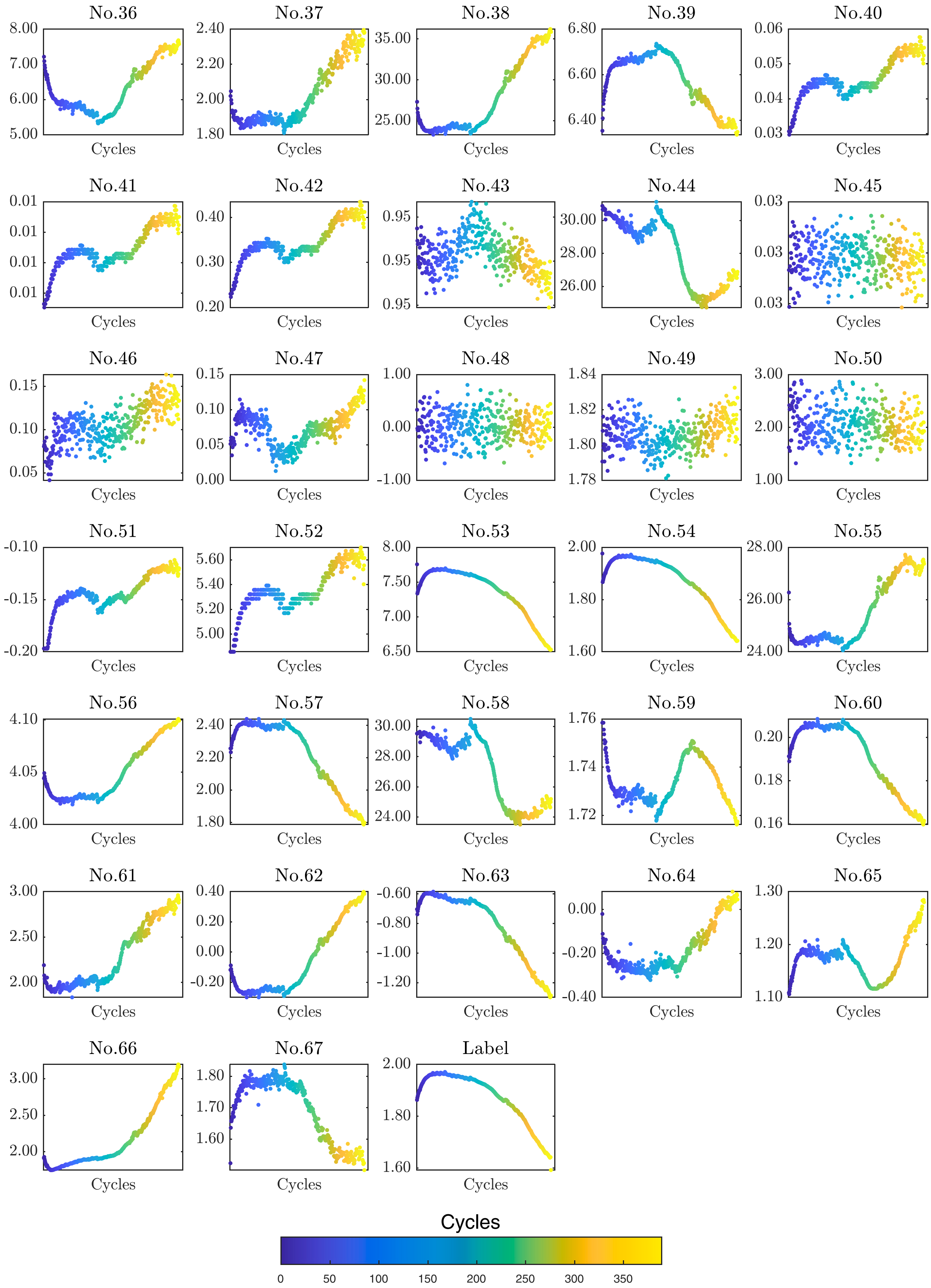


Fig. B.15. A visualization of partial handcraft features for batch 1 battery 1 in XJTU battery dataset. Subheadings No. 36 to No. 67 correspond to Feature No. in Table B.7. The last subplot shows the label (capacity).

Table B.7
Handcraft features from charging process. Note: CC represents constant current, CV represents constant voltage, CCCV represents constant current charge voltage, CVCC represents constant voltage charge current, CCCT denotes constant current charge time and CVCT denotes constant voltage charge time.

Data type	Feature no.	Feature
charge	1	Cumulated charge energy
	2	Cumulated charge capacity
	3	Mean charge time
	4	Mean charge voltage
	5	Mean charge current
	6	Mean charge temperature
	7	Standard deviation of charge current
	8	Standard deviation of charge voltage
	9	Standard deviation of charge temperature
	10	skewness of charge current
	11	skewness of charge voltage
	12	skewness of charge temperature
	13	kurtosis of charge current
	14	kurtosis of charge voltage
	15	kurtosis of charge temperature
CC mode	16	Energy during CC segment
	17	Capacity during CC segment
	18	Mean time during CC segment
	19	Mean voltage during CC segment
	20	Mean temperature during CC segment
	21	Standard deviation of voltage during CC segment
	22	Standard deviation of temperature during CC segment
	23	skewness of voltage during CC segment
	24	skewness of temperature during CC segment
	25	kurtosis of voltage during CC segment
	26	kurtosis of temperature during CC segment
	27	constant current charge time
	28	Entropy of CCCV-CCCT curve
	29	Energy during 3.9–4.0 V
	30	Capacity during 3.9–4.0 V
	31	Mean time during 3.9–4.0 V
	32	Mean voltage during 3.9–4.0 V
	33	Mean temperature during 3.9–4.0 V
	34	Standard deviation of voltage during 3.9–4.0 V
	35	Standard deviation of temperature during 3.9–4.0 V
	36	skewness of voltage during 3.9–4.0 V
	37	skewness of temperature during 3.9–4.0 V
	38	kurtosis of voltage during 3.9–4.0 V
	39	kurtosis of temperature during 3.9–4.0 V
	40	Slope of CCCV-CCCT curve during 3.9–4.0 V
	41	Entropy of CCCV-CCCT curve during 3.9–4.0 V
CV mode	42	Energy during CV segment
	43	Capacity during CV segment
	44	Mean time during CV segment
	45	Mean current during CV segment
	46	Mean temperature during CV segment
	47	Standard deviation of current during CV segment
	48	Standard deviation of temperature during CV segment
	49	skewness of current during CV segment
	50	skewness of temperature during CV segment
	51	kurtosis of current during CV segment
	52	kurtosis of temperature during CV segment
	53	constant voltage charge time
	54	Entropy of CVCC-CVCT curve
	55	Energy during 1.0–0.9 A
	56	Capacity during 1.0–0.9 A
	57	Mean time during 1.0–0.9 A
	58	Mean current during 1.0–0.9 A
	59	Mean temperature during 1.0–0.9 A
	60	Standard deviation of current during 1.0–0.9 A
	61	Standard deviation of temperature during 1.0–0.9 A
	62	skewness of current during 1.0–0.9 A
	63	skewness of temperature during 1.0–0.9 A
	64	kurtosis of current during 1.0–0.9 A
	65	kurtosis of temperature during 1.0–0.9 A
	66	slope of CVCC-CVCT curve during 1.0–0.9 A
	67	Entropy of CVCC-CVCT curve during 1.0–0.9 A

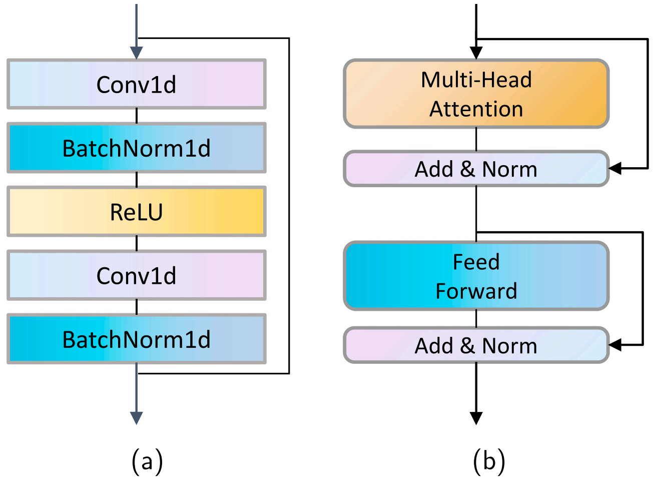


Fig. C.16. **(a)**: Details of ResBlock1d in CNN model, which is exactly the same as that in 1D ResNet. **(b)**: Details of Encoder in Attention model, which are similar to TransformerEncoder.

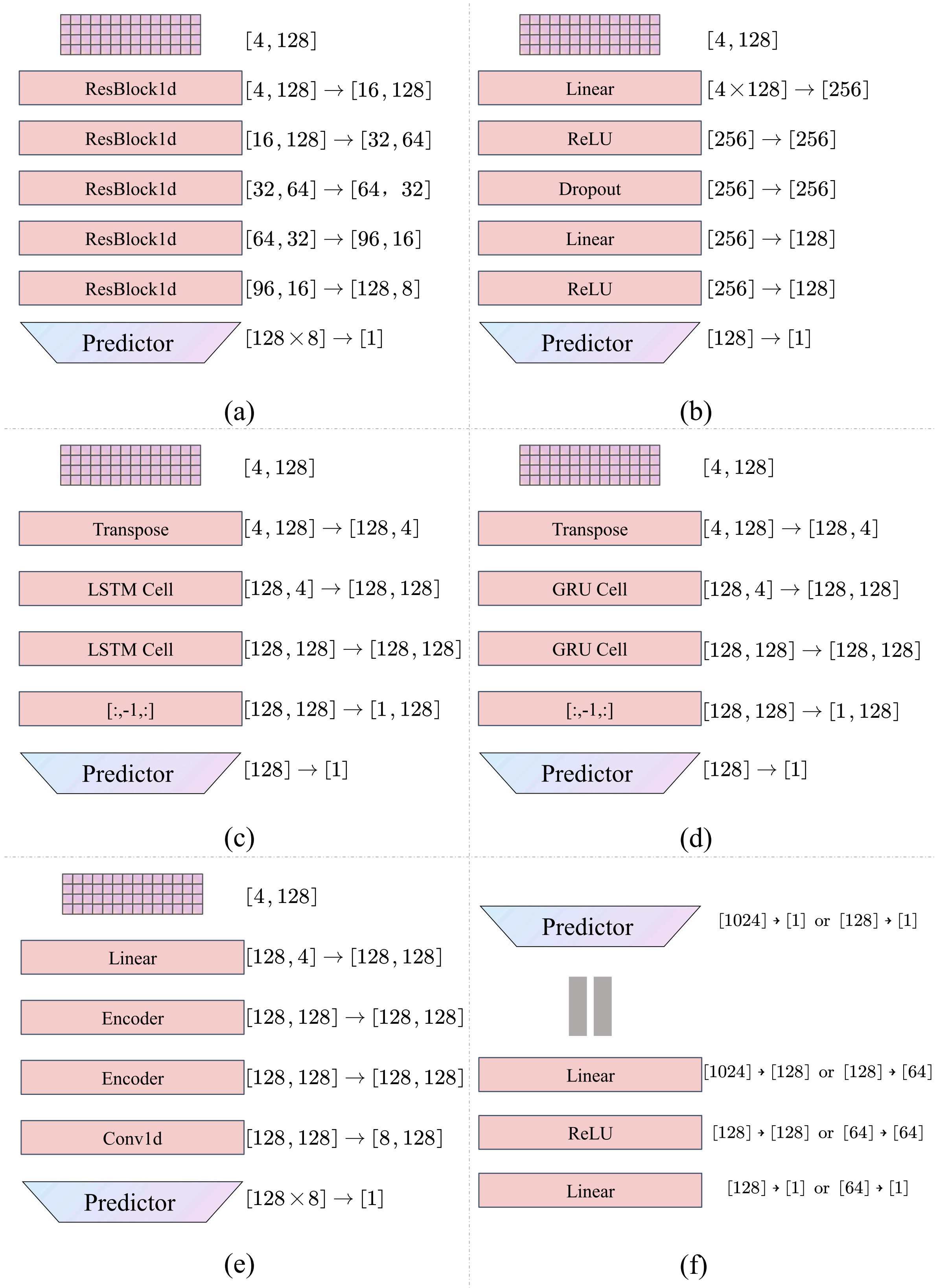


Fig. C.17. A visualization of model structures. **(a).** CNN. **(b).** MLP. **(c).** LSTM. **(d).** GRU. **(e).** Attention. **(f).** Details of the Predictor.

Table D.8
The results of XJTU battery dataset with [−1,1] normalization. For each **column**, within each batch, the best result of three input types was **bolded**. For each **row**, the best result among five models was shown in *italics*.

Model		CNN			LSTM			GRU			MLP			Attention		
Input type	Batch	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE
A	1	12.362	13.421	0.300	20.080	22.542	0.788	21.891	24.456	0.853	*7.398*	*8.121*	*0.098*	9.818	10.784	0.165
	2	18.012	19.338	0.780	21.697	23.795	0.789	23.102	25.323	0.845	*11.910*	*12.910*	*0.257*	12.489	13.574	0.282
	3	7.815	8.474	0.108	26.675	29.405	1.337	28.004	30.621	1.278	*7.261*	*7.907*	*0.096*	8.100	8.742	0.107
	4	*8.494*	*9.240*	*0.127*	23.775	26.404	1.142	25.403	28.080	1.189	9.585	10.437	0.166	9.893	10.742	0.173
	5	14.928	16.230	0.431	36.014	40.298	2.133	36.115	40.443	2.165	***12.363***	***13.578***	***0.271***	12.741	13.990	0.288
	6	*18.795*	*20.374*	*0.723*	28.440	31.280	1.512	28.078	30.915	1.503	23.320	25.331	1.157	21.913	23.822	1.030
B	1	*6.235*	*6.822*	0.070	7.480	8.169	0.083	7.834	8.487	0.088	6.408	6.960	*0.062*	7.803	8.449	0.095
	2	**10.766**	**11.590**	**0.251**	10.711	11.692	0.191	10.715	11.626	0.189	***6.186***	***6.675***	***0.067***	10.791	11.806	0.205
	3	6.509	6.898	0.072	6.343	6.766	0.063	6.290	6.683	0.060	*5.562*	*5.930*	*0.053*	8.705	9.262	0.136
	4	*7.981*	*8.695*	0.122	10.894	11.863	0.191	11.012	11.949	0.184	8.098	8.783	*0.108*	9.712	10.500	0.158
	5	16.668	18.079	0.487	35.771	40.082	2.138	36.165	40.443	2.133	16.901	18.571	0.497	*14.103*	*15.475*	*0.325*
	6	*16.882*	*18.370*	*0.543*	33.128	36.262	1.890	33.045	36.218	1.900	24.581	26.849	1.054	20.517	22.356	0.759
C	1	**5.816**	**6.311**	**0.060**	**4.581**	**5.039**	**0.039**	***3.862***	***4.223***	***0.028***	**4.254**	**4.612**	**0.034**	**6.606**	**7.238**	**0.078**
	2	11.466	12.293	0.294	**9.337**	**10.105**	**0.135**	**8.149**	**8.791**	**0.109**	*7.538*	*8.092*	*0.104*	**10.474**	**11.414**	**0.171**
	3	**4.783**	**5.081**	**0.042**	**4.726**	**5.056**	**0.039**	**3.938**	**4.194**	**0.026**	***3.831***	***4.074***	***0.024***	**5.914**	**6.332**	**0.058**
	4	**5.350**	**5.781**	**0.045**	**5.954**	**6.456**	**0.057**	**5.399**	**5.831**	**0.045**	***4.945***	***5.324***	***0.038***	**6.264**	**6.754**	**0.065**
	5	**11.553**	**12.643**	**0.237**	**9.641**	**10.663**	**0.172**	***9.409***	***10.382***	***0.154***	15.289	16.641	0.451	**10.225**	**11.244**	**0.180**
	6	***10.048***	***10.841***	***0.171***	**11.992**	**13.065**	**0.248**	**11.260**	**12.241**	**0.218**	**11.237**	**12.196**	**0.214**	**11.403**	**12.357**	**0.227**

* A is the charging data; B is the partial charging data; C is the handcraft features.
Note: (1). All values are the average value of 3 experiments; (2). For intuitive display, all values have been magnified 1000 times.

Table D.9
The results of XJTU battery dataset with [0,1] normalization. For each **column**, within each batch, the best result of three input types was **bolded**. For each **row**, the best result among five models was shown in *italics*.

Model		CNN			LSTM			GRU			MLP			Attention		
Input type	Batch	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE
A	1	10.431	11.438	0.198	22.463	25.174	0.934	23.863	26.676	1.007	*8.134*	*8.964*	*0.121*	8.779	9.594	0.133
	2	16.782	18.137	0.743	25.637	28.138	0.975	24.853	27.307	0.945	13.246	14.352	0.293	*12.640*	*13.747*	*0.268*
	3	9.786	10.548	0.187	27.089	29.875	1.360	28.528	31.516	1.502	*7.696*	*8.427*	*0.112*	7.950	8.675	0.118
	4	***7.560***	***8.233***	***0.103***	25.121	27.875	1.216	25.150	27.902	1.221	10.124	11.042	0.201	10.083	10.988	0.191
	5	16.127	17.388	0.466	35.953	40.263	2.142	36.056	40.359	2.143	*14.737*	*16.164*	*0.381*	17.148	18.802	0.487
	6	*17.672*	*19.194*	*0.679*	28.433	31.287	1.514	27.831	30.671	1.497	22.340	24.312	1.001	20.747	22.587	0.919
B	1	***5.703***	***6.288***	***0.058***	8.852	9.758	0.130	8.371	9.200	0.114	8.549	9.274	0.108	12.275	13.539	0.265
	2	**13.238**	**14.201**	**0.463**	11.769	12.921	0.238	11.250	12.310	0.211	*9.635*	*10.503*	*0.154*	11.752	12.858	0.242
	3	***5.710***	***6.099***	***0.056***	7.192	7.723	0.082	6.932	7.449	0.077	6.858	7.305	0.075	8.436	9.066	0.130
	4	*8.087*	*8.810*	*0.118*	11.593	12.682	0.225	11.541	12.583	0.213	9.923	10.759	0.151	10.770	11.688	0.196
	5	**15.704**	**17.161**	**0.432**	35.621	39.928	2.141	35.905	40.214	2.139	*14.824*	*16.268*	*0.356*	17.433	19.204	0.520
	6	*17.409*	*18.972*	*0.567*	33.432	36.537	1.905	32.969	36.111	1.889	27.453	30.101	1.330	29.126	31.901	1.521
C	1	27.359	29.201	1.459	**5.210**	**5.762**	**0.051**	***4.299***	***4.693***	***0.032***	**6.754**	**7.284**	**0.082**	**6.767**	**7.338**	**0.069**
	2	28.032	30.128	1.581	**10.023**	**10.862**	**0.156**	***8.777***	***9.488***	***0.120***	**9.040**	**9.743**	**0.142**	**10.148**	**11.005**	**0.163**
	3	10.320	10.928	0.418	**5.406**	**5.802**	**0.052**	***4.371***	***4.656***	***0.033***	**4.486**	**4.765**	**0.034**	**5.689**	**6.050**	**0.054**
	4	8.160	8.739	0.162	**6.546**	**7.122**	**0.073**	**5.777**	**6.235**	**0.053**	***5.492***	***5.913***	***0.048***	**6.015**	**6.499**	**0.059**
	5	34.582	37.306	1.752	**9.911**	**10.987**	**0.189**	***9.873***	***10.899***	***0.173***	**13.836**	**15.129**	**0.317**	**9.940**	**11.020**	**0.201**
	6	***9.702***	***10.485***	***0.176***	**17.241**	**18.768**	**0.663**	**11.182**	**12.195**	**0.225**	**10.830**	**11.801**	**0.210**	**10.239**	**11.126**	**0.192**

* A is the charging data; B is the partial charging data; C is the handcraft features.
Note: (1). All values are the average value of 3 experiments; (2). For intuitive display, all values have been magnified 1000 times.

Table D.10
The results of XJTU battery dataset with z-score normalization. For each **column**, within each batch, the best result of three input types was **bolded**. For each **row**, the best result among five models was shown in *italics*.

Model		CNN			LSTM			GRU			MLP			Attention		
Input type	Batch	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE
A	1	21.797	23.585	0.877	20.702	23.178	0.820	23.798	26.636	1.025	**8.187**	**8.840**	**0.102**	***6.715***	***7.358***	***0.093***
	2	20.930	22.461	1.115	26.535	29.126	1.142	31.294	34.319	1.512	**17.556**	**18.912**	**0.463**	*12.179*	*13.219*	*0.322*
	3	7.229	7.710	0.146	27.446	30.440	1.510	31.143	34.483	1.811	**11.234**	**12.050**	**0.207**	***6.619***	***7.162***	***0.081***
	4	*7.816*	*8.420*	*0.116*	24.120	26.791	1.185	25.632	28.433	1.275	**8.683**	**9.391**	**0.128**	8.012	8.750	0.121
	5	22.799	24.774	0.847	36.042	40.334	2.135	35.762	40.078	2.139	**13.419**	**14.764**	**0.329**	*11.714*	***12.903***	***0.279***
	6	*12.776*	*13.803*	*0.308*	28.277	31.121	1.494	29.232	32.103	1.554	18.522	20.192	0.705	16.304	17.629	0.515
B	1	*36.364*	*40.309*	2.474	37.388	41.412	*2.438*	44.055	48.648	2.776	287.701	311.225	143.892	43.185	47.013	3.072
	2	41.711	45.850	2.975	35.607	39.325	*2.153*	*35.514*	*39.264*	2.303	278.173	301.800	186.625	55.613	60.714	5.971
	3	42.911	46.730	3.077	*42.131*	*46.047*	*2.845*	43.967	47.871	2.966	178.370	188.504	59.169	47.318	51.557	3.766
	4	38.165	41.769	2.454	*35.953*	*39.366*	*2.218*	36.219	39.652	2.339	191.194	203.646	75.885	38.893	42.401	2.494
	5	24.789	26.920	1.036	36.250	40.549	2.151	36.205	40.504	2.148	20.583	22.642	0.709	***11.708***	*12.987*	*0.296*
	6	*16.264*	*17.745*	*0.501*	27.187	29.926	1.457	33.192	36.314	1.890	25.326	27.738	1.157	17.437	19.050	0.605
C	1	***5.249***	***5.732***	***0.047***	**8.294**	**9.029**	**0.174**	**8.894**	**9.729**	**0.210**	42.566	45.919	3.260	7.203	7.839	0.116
	2	***8.706***	***9.434***	***0.149***	**10.202**	**11.053**	**0.189**	**14.835**	**16.062**	**0.484**	21.566	23.213	0.935	**11.195**	**12.133**	**0.231**
	3	***5.102***	***5.545***	***0.051***	**6.926**	**7.487**	**0.108**	**11.205**	**12.152**	**0.358**	26.012	27.683	1.172	7.406	8.023	0.113
	4	***5.951***	***6.479***	***0.060***	**6.326**	**6.866**	**0.065**	**11.605**	**12.605**	**0.320**	13.359	14.365	0.380	**7.509**	**8.175**	**0.100**
	5	**14.095**	**15.552**	**0.462**	***13.387***	***14.853***	**0.429**	**16.416**	**18.183**	**0.600**	48.896	53.616	3.917	14.254	15.698	*0.399*
	6	***10.613***	***11.466***	***0.206***	**12.916**	**13.930**	**0.324**	**13.673**	**14.733**	**0.368**	**16.041**	**17.145**	**0.501**	**11.331**	**12.245**	**0.236**

* A is the charging data; B is the partial charging data; C is the handcraft features.
Note: (1). All values are the average value of 3 experiments; (2). For intuitive display, all values have been magnified 1000 times.

Table D.11
The results of Toyota-MIT-Stanford dataset with [−1,1] normalization. For each **column**, within each batch, the best result of three input types was **bolded**. For each **row**, the best result among five models was shown in *italics*.

Model		CNN			LSTM			GRU			MLP			Attention		
Input type	Group	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE
A	1	17.439	18.865	0.825	12.057	13.446	0.352	12.167	13.515	0.335	11.349	12.572	0.259	*8.666*	*9.659*	*0.197*
	2	28.806	30.787	1.466	21.895	23.840	0.768	22.216	24.125	0.754	15.505	16.823	0.458	*11.476*	*12.511*	*0.263*
	3	12.529	13.549	0.513	***9.524***	***10.579***	0.210	**9.567**	**10.597**	*0.200*	14.133	15.538	0.434	11.331	12.521	0.294
	4	**8.893**	**9.546**	**0.189**	**8.523**	**9.414**	0.181	8.746	9.626	0.177	6.980	7.724	0.138	***6.213***	***6.855***	***0.095***
	5	46.447	49.840	4.016	34.483	37.625	2.410	42.620	46.191	3.933	22.823	24.710	0.888	*17.656*	*19.138*	*0.607*
	6	9.772	10.572	0.341	8.657	9.464	0.173	9.006	9.786	0.166	9.530	10.375	0.198	*7.543*	*8.244*	*0.139*
	7	*8.538*	*9.251*	***0.101***	12.321	13.518	0.257	12.225	13.359	0.246	13.884	15.186	0.344	10.484	11.470	0.190
	8	14.233	15.220	0.574	12.979	14.242	0.292	12.255	13.439	0.265	14.525	15.820	0.374	*11.952*	*13.028*	*0.258*
	9	18.984	20.433	0.916	12.362	13.663	0.352	11.764	12.966	0.308	9.917	11.014	0.291	*9.424*	*10.450*	*0.262*
B	1	10.728	11.597	0.287	10.223	10.996	0.160	10.381	11.236	0.172	**5.364**	**5.822**	0.067	***5.078***	***5.524***	***0.055***
	2	52.206	55.765	3.553	12.513	13.592	0.288	12.905	14.022	0.307	10.989	11.871	0.257	*9.578*	*10.411*	*0.182*
	3	28.259	30.263	1.585	10.527	11.494	0.221	12.285	13.433	0.277	***7.773***	***8.489***	***0.133***	8.896	9.649	**0.140**
	4	16.096	17.217	0.594	8.879	9.707	**0.145**	**8.488**	**9.357**	**0.158**	***5.931***	***6.527***	***0.073***	6.553	7.214	0.105
	5	17.604	19.028	0.699	**6.905**	**7.684**	0.135	**6.758**	**7.607**	0.154	***5.300***	***5.865***	***0.062***	**5.957**	**6.505**	**0.078**
	6	32.939	35.311	1.969	7.358	8.050	0.110	**6.394**	7.028	0.096	**5.938**	**6.462**	0.099	***5.360***	***5.870***	***0.066***
	7	47.674	50.832	3.848	23.928	25.932	1.080	33.101	35.670	2.545	***10.357***	***11.162***	**0.271**	10.649	11.458	*0.218*
	8	**8.332**	**9.076**	**0.158**	**8.757**	**9.561**	**0.124**	**8.409**	**9.207**	**0.130**	***4.370***	***4.859***	***0.063***	**9.330**	**10.161**	**0.143**
	9	20.065	21.618	0.809	**6.910**	**7.602**	0.110	**7.198**	**7.950**	0.126	***6.756***	***7.382***	**0.125**	7.024	7.668	*0.090*
C	1	***5.748***	***6.228***	***0.056***	**6.664**	**7.213**	**0.066**	**6.109**	**6.584**	***0.056***	6.842	7.355	**0.066**	7.314	7.958	0.086
	2	**6.857**	**7.338**	**0.082**	**7.980**	**8.567**	**0.098**	**7.426**	**7.972**	**0.079**	***6.574***	***7.080***	***0.067***	**8.525**	**9.168**	**0.123**
	3	**9.859**	**10.562**	***0.139***	10.375	11.181	**0.155**	10.874	11.701	**0.164**	14.111	15.116	0.260	***8.879***	***9.616***	0.143
	4	19.982	21.262	0.648	13.130	14.070	0.237	13.267	14.176	*0.208*	26.294	27.892	1.289	*12.515*	*13.430*	0.227
	5	**8.298**	**8.966**	**0.129**	8.777	9.439	**0.110**	*7.572*	*8.124*	***0.081***	9.739	10.416	0.134	10.461	11.295	0.161
	6	**8.709**	**9.276**	**0.105**	**7.019**	**7.479**	**0.080**	*6.506*	***6.904***	***0.070***	7.027	7.464	**0.075**	7.339	7.879	0.089
	7	**8.150**	**8.870**	0.118	**5.496**	**5.997**	**0.056**	***5.138***	***5.595***	***0.050***	10.785	11.696	0.300	**7.925**	**8.654**	**0.099**
	8	12.468	13.371	0.219	12.001	12.880	0.187	12.526	13.410	0.205	11.823	12.638	0.197	*10.393*	*11.203*	*0.154*
	9	**9.335**	**10.018**	**0.143**	7.333	7.898	**0.079**	7.640	8.205	**0.082**	20.148	21.440	0.646	***6.583***	***7.145***	***0.070***

* A is the charging data; B is the partial charging data; C is the handcraft features.
Note: (1). All values are the average value of 3 experiments; (2). For intuitive display, all values have been magnified 1000 times.

Table D.12
The results of Toyota-MIT-Stanford dataset with [0,1] normalization. For each **column**, within each batch, the best result of three input types was **bolded**. For each **row**, the best result among five models was shown in *italics*.

Model		CNN			LSTM			GRU			MLP			Attention		
Input type	Group	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE
A	1	22.775	24.579	1.249	11.601	13.057	0.376	12.482	13.911	0.359	10.385	11.597	*0.259*	*9.650*	*10.847*	0.274
	2	31.741	33.796	2.047	23.943	26.333	1.123	21.846	23.851	0.771	*17.176*	*18.627*	*0.470*	17.826	19.586	0.618
	3	**12.557**	**13.629**	**0.372**	10.460	11.637	0.256	***9.644***	***10.712***	*0.215*	**11.999**	**13.237**	**0.274**	9.882	10.996	0.229
	4	***6.175***	***6.691***	***0.082***	**8.880**	**9.869**	0.211	8.614	9.524	0.186	**6.982**	**7.717**	0.127	7.541	8.344	0.162
	5	31.019	33.359	2.241	21.047	23.284	0.939	33.437	36.408	2.037	33.563	36.204	2.254	*17.730*	*19.466*	*0.635*
	6	**14.363**	**15.549**	**0.570**	9.758	10.726	0.238	9.068	9.917	0.188	8.214	8.976	0.165	*6.835*	*7.576*	*0.162*
	7	**8.955**	**9.760**	***0.137***	12.749	14.070	0.302	12.019	13.213	0.262	**12.255**	**13.392**	**0.261**	*8.400*	*9.304*	0.165
	8	**16.541**	**17.768**	**0.599**	13.533	14.890	0.335	12.775	14.041	0.298	12.060	13.198	0.264	*8.756*	*9.689*	*0.195*
	9	17.481	18.882	0.971	14.823	16.392	0.485	12.147	13.455	0.340	*10.887*	*12.006*	*0.339*	11.577	12.862	0.346
B	1	15.891	16.899	0.754	11.117	11.956	0.190	10.732	11.551	0.161	***6.514***	***7.063***	**0.086**	**6.710**	**7.263**	***0.081***
	2	53.956	57.431	3.809	12.517	13.615	0.282	12.932	14.008	0.267	13.038	14.134	0.281	*11.284*	*12.264*	*0.207*
	3	18.314	19.708	0.766	10.591	11.582	0.237	11.049	12.048	0.210	12.307	13.361	0.313	***8.342***	***9.056***	*0.125*
	4	21.842	23.502	1.206	9.434	10.283	**0.157**	**8.562**	**9.368**	**0.135**	7.426	8.155	**0.121**	***6.687***	***7.317***	***0.100***
	5	**21.767**	**23.437**	**1.202**	**7.616**	**8.467**	0.158	**6.661**	7.438	0.119	**5.649**	**6.251**	***0.066***	***5.020***	***5.608***	**0.076**
	6	15.222	16.288	0.766	7.809	8.562	0.132	**6.443**	**7.048**	0.088	**5.657**	**6.184**	**0.084**	***4.676***	***5.173***	***0.065***
	7	37.863	40.576	2.364	23.751	25.845	0.972	35.132	37.852	2.662	35.692	38.187	2.994	*11.331*	*12.424*	*0.286*
	8	24.177	26.090	1.465	**9.346**	**10.217**	**0.145**	**8.672**	**9.475**	**0.121**	***5.346***	***5.914***	***0.080***	**7.017**	**7.649**	**0.093**
	9	**14.945**	**16.066**	**0.761**	**7.448**	**8.182**	0.123	7.086	7.788	0.107	**7.161**	**7.840**	**0.115**	***5.839***	***6.448***	*0.093*
C	1	**10.775**	**11.622**	**0.343**	**7.079**	**7.693**	**0.075**	***6.556***	***7.077***	***0.061***	9.370	10.121	0.135	6.972	7.593	0.082
	2	**9.439**	**10.142**	**0.197**	**8.989**	**9.681**	**0.127**	**9.142**	**9.798**	**0.125**	**11.839**	**12.686**	**0.214**	***6.717***	***7.225***	***0.081***
	3	19.355	20.757	0.786	**10.382**	**11.200**	**0.159**	10.574	11.379	**0.163**	16.402	17.605	0.350	*8.636*	*9.344*	***0.124***
	4	56.621	60.226	3.762	12.931	13.877	0.239	13.247	14.160	*0.226*	22.176	23.546	0.681	*12.273*	*13.192*	0.235
	5	35.065	37.486	2.108	9.128	9.843	**0.118**	*6.849*	***7.371***	***0.074***	18.213	19.545	0.448	8.306	8.962	0.103
	6	17.866	18.965	0.729	**7.670**	**8.206**	**0.094**	*7.065*	*7.513*	***0.075***	10.223	10.927	0.148	7.426	7.940	0.087
	7	23.780	25.773	1.034	***5.961***	***6.545***	**0.070**	**6.268**	**6.808**	***0.067***	14.122	15.281	0.275	**7.137**	**7.800**	**0.087**
	8	22.843	24.469	0.805	*11.835*	*12.743*	*0.186*	12.162	13.035	0.192	13.067	14.038	0.255	12.909	13.878	0.211
	9	39.385	42.106	1.928	7.619	8.231	**0.093**	***6.448***	***6.957***	***0.073***	28.892	30.912	1.252	7.404	8.006	**0.081**

* A is the charging data; B is the partial charging data; C is the handcraft features.

Note: (1). All values are the average value of 3 experiments; (2). For intuitive display, all values have been magnified 1000 times.

Table D.13
The results of Toyota-MIT-Stanford dataset with z-score normalization. For each **column**, within each batch, the best result of three input types was **bolded**. For each **row**, the best result among five models was shown in *italics*.

Model		CNN			LSTM			GRU			MLP			Attention		
Input type	Group	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE	MAE	MAPE	MSE
A	1	14.054	15.271	0.452	28.953	32.137	1.447	32.149	35.394	1.684	***5.415***	***5.935***	***0.072***	12.137	13.169	0.335
	2	26.371	28.190	1.140	24.371	26.872	1.238	23.683	26.138	1.191	***17.249***	***18.555***	***0.455***	20.198	21.751	0.688
	3	19.135	20.719	0.788	27.414	30.333	1.304	27.343	30.347	1.346	***7.665***	***8.409***	***0.123***	**11.078**	12.008	0.269
	4	15.994	17.167	0.622	19.086	20.850	0.682	29.487	32.170	1.302	**14.595**	**15.641**	**0.398**	***9.373***	***10.146***	***0.189***
	5	32.678	34.986	1.919	22.437	24.787	1.030	21.598	23.889	0.983	16.392	17.554	*0.412*	*14.380*	*15.613*	0.526
	6	18.445	19.693	0.676	25.916	28.381	1.291	25.151	27.629	1.244	***7.284***	***7.900***	***0.126***	8.868	9.535	0.158
	7	*10.617*	*11.462*	*0.236*	30.985	33.827	1.824	34.600	37.633	2.264	**21.274**	**22.915**	1.005	15.784	17.169	0.473
	8	14.474	15.645	0.492	24.997	27.366	1.055	34.568	37.613	1.880	***6.774***	***7.423***	***0.107***	15.266	16.625	0.457
	9	18.928	20.477	0.734	25.457	28.164	1.280	25.785	28.538	1.299	***7.813***	*8.606*	*0.164*	11.996	12.990	0.258
B	1	16.653	17.781	0.616	12.139	13.031	0.218	11.802	12.688	***0.213***	*6.882*	*7.403*	*0.093*	8.666	9.291	0.156
	2	*20.808*	*22.316*	*0.645*	24.147	25.767	0.920	22.313	23.912	0.757	23.058	24.937	1.058	34.659	37.325	2.234
	3	17.473	18.839	0.683	12.076	12.927	0.238	***8.538***	***9.160***	***0.109***	10.671	11.557	0.179	16.089	17.471	0.509
	4	22.815	24.445	0.968	**8.578**	**9.325**	**0.130**	***8.421***	***9.182***	***0.127***	14.867	16.204	0.502	9.934	10.796	0.195
	5	12.814	13.937	0.344	***6.810***	***7.442***	**0.079**	**7.207**	***7.797***	**0.075**	**10.403**	**11.283**	**0.178**	**11.340**	12.378	0.272
	6	21.942	23.693	0.960	11.165	12.055	**0.192**	**10.123**	**10.965**	**0.165**	*7.747*	*8.361*	*0.143*	11.478	12.366	0.320
	7	24.968	26.794	1.419	11.962	12.884	*0.229*	***10.696***	***11.599***	***0.196***	23.187	24.999	**0.907**	19.007	20.428	0.830
	8	15.131	16.447	0.451	17.279	18.606	0.494	15.194	16.367	0.366	*8.711*	*9.459*	*0.140*	10.069	10.918	0.198
	9	16.830	18.267	0.839	**8.779**	**9.532**	**0.119**	11.026	11.898	0.174	*7.911*	***8.588***	***0.105***	9.734	10.507	0.324
C	1	***6.072***	***6.554***	***0.064***	**6.455**	**6.957**	**0.067**	**10.940**	**11.891**	0.239	53.601	58.041	6.204	**7.951**	**8.607**	**0.119**
	2	***6.768***	***7.251***	***0.081***	**8.976**	**9.600**	**0.176**	**11.140**	**11.970**	**0.299**	87.473	93.580	23.135	**8.869**	**9.496**	**0.175**
	3	**11.559**	**12.510**	**0.205**	**11.146**	**12.080**	***0.190***	13.716	14.826	0.346	98.937	106.096	17.034	*11.088*	***11.961***	**0.204**
	4	***12.553***	***13.422***	***0.204***	13.660	14.625	0.269	13.120	14.078	0.265	76.932	81.989	11.607	13.066	13.969	0.235
	5	***8.785***	***9.421***	***0.120***	12.478	13.409	0.402	12.672	13.638	0.240	61.801	66.524	6.658	11.347	**12.163**	**0.195**
	6	***6.400***	***6.836***	***0.067***	**10.854**	**11.570**	0.383	11.167	11.915	0.366	35.105	37.564	2.069	**8.273**	**8.850**	**0.114**
	7	***8.218***	***9.011***	**0.122**	**8.355**	**9.186**	***0.110***	14.404	15.708	0.648	40.399	43.933	3.538	**11.269**	**12.304**	**0.290**
	8	***6.415***	***6.903***	***0.068***	**9.274**	**10.002**	**0.143**	**9.622**	**10.370**	**0.154**	37.891	40.835	2.342	**8.638**	**9.318**	**0.121**
	9	***7.152***	***7.678***	***0.077***	8.881	9.571	0.126	**9.505**	**10.234**	**0.161**	60.913	65.133	6.490	**7.525**	**8.067**	**0.090**

* A is the charging data; B is the partial charging data; C is the handcraft features.

Note: (1). All values are the average value of 3 experiments; (2). For intuitive display, all values have been magnified 1000 times.

References

[1] R. Schmuch, R. Wagner, G. Hörpel, T. Placke, M. Winter, Performance and cost of materials for lithium-based rechargeable automotive batteries, Nature Energy 3 (4) (2018) 267–278.

[2] Y. Liang, C.-Z. Zhao, H. Yuan, Y. Chen, W. Zhang, J.-Q. Huang, D. Yu, Y. Liu, M.-M. Titirici, Y.-L. Chueh, et al., A review of rechargeable batteries for portable electronic devices, InfoMat 1 (1) (2019) 6–32.

[3] G. Harper, R. Sommerville, E. Kendrick, L. Driscoll, P. Slater, R. Stolkin, A. Walton, P. Christensen, O. Heidrich, S. Lambert, et al., Recycling lithium-ion batteries from electric vehicles, Nature 575 (7781) (2019) 75–86.

[4] R. Xiong, J. Kim, W. Shen, C. Lv, H. Li, X. Zhu, W. Zhao, B. Gao, H. Guo, C. Zhang, et al., Key technologies for electric vehicles, Green Energy Intell. Transp. 1 (2) (2022).

[5] E. Almaita, S. Alshkoor, E. Abdelsalam, F. Almomani, State of charge estimation for a group of lithium-ion batteries using long short-term memory neural network, J. Energy Storage 52 (2022) 104761.

[6] D. Zhao, Z. Zhou, P. Zhang, Y. Zhang, Z. Feng, Y. Yang, Y. Cao, Health condition assessment of satellite li-ion battery pack considering battery inconsistency and pack performance indicators, J. Energy Storage 60 (2023) 106604.

[7] F. Sun, Green energy and intelligent transportation—promoting green and intelligent mobility, Green Energy Intell. Transp. 1 (1) (2022) 100017.

[8] D. Ren, H. Hsu, R. Li, X. Feng, D. Guo, X. Han, L. Lu, X. He, S. Gao, J. Hou, et al., A comparative investigation of aging effects on thermal runaway behavior of lithium-ion batteries, ETransportation 2 (2019) 100034.

[9] H. He, F. Sun, Z. Wang, C. Lin, C. Zhang, R. Xiong, J. Deng, X. Zhu, P. Xie, S. Zhang, et al., China's battery electric vehicles lead the world: Achievements in technology system architecture and technological breakthroughs, Green Energy Intell. Transp. (2022) 100020.

[10] H. Rauf, M. Khalid, N. Arshad, Machine learning in state of health and remaining useful life estimation: Theoretical and technological development in battery degradation modelling, Renew. Sustain. Energy Rev. 156 (2022) 111903.

[11] Y. Zhang, Y.-F. Li, Prognostics and health management of lithium-ion battery using deep learning methods: A review, Renew. Sustain. Energy Rev. 161 (2022) 112282.

[12] M. Daigle, C.S. Kulkarni, Electrochemistry-based battery modeling for prognostics, in: Annual Conference of the PHM Society, Vol. 5, No. 1, 2013.

[13] M. Daigle, S. Sankararaman, Advanced methods for determining prediction uncertainty in model-based prognostics with application to planetary rovers, in: Annual Conference of the PHM Society, Vol. 5, No. 1, 2013.

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

[15] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Commun. ACM 60 (6) (2017) 84–90.

[16] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, et al., A survey on vision transformer, IEEE Trans. Pattern Anal. Mach. Intell. 45 (1) (2022) 87–110.

[17] J. Hirschberg, C.D. Manning, Advances in natural language processing, Science 349 (6245) (2015) 261–266.

[18] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, et al., Improving Language Understanding by Generative Pre-Training, OpenAI, 2018.

[19] C. Luo, X. Yang, A. Yuille, Self-supervised pillar motion learning for autonomous driving, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2021, pp. 3183–3192.

[20] A. Prakash, K. Chitta, A. Geiger, Multi-modal fusion transformer for end-to-end autonomous driving, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2021, pp. 7077–7087.

[21] Z. Zhao, T. Li, J. Wu, C. Sun, S. Wang, R. Yan, X. Chen, Deep learning algorithms for rotating machinery intelligent diagnosis: An open source benchmark study, ISA Trans. 107 (2020) 224–255.

[22] Z. Zhai, J. Wen, F. Wang, Z. Zhao, Y. Guo, X. Chen, Remaining useful life prediction of aero-engine based on transformer with tendency retainment, in: 2022 International Conference on Sensing, Measurement & Data Analytics in the Era of Artificial Intelligence, ICSMD, IEEE, 2022, pp. 1–6.

[23] Z. Zhai, Y. Di, F. Wang, Z. Leng, Z. Zhao, X. Chen, Adversarially-trained graph convolutional recurrent autoencoder for spacecraft anomaly detection with missing data, in: 2022 International Conference on Sensing, Measurement & Data Analytics in the Era of Artificial Intelligence, ICSMD, IEEE, 2022, pp. 1–7.

[24] K. Luo, X. Chen, H. Zheng, Z. Shi, A review of deep learning approach to predicting the state of health and state of charge of lithium-ion batteries, J. Energy Chem. (2022).

[25] H. Meng, Y.-F. Li, A review on prognostics and health management (PHM) methods of lithium-ion batteries, Renew. Sustain. Energy Rev. 116 (2019) 109405.

[26] Y. Liu, Y. He, H. Bian, W. Guo, X. Zhang, A review of lithium-ion battery state of charge estimation based on deep learning: Directions for improvement and future trends, J. Energy Storage 52 (2022) 104664.

[27] P.M. Attia, A. Grover, N. Jin, K.A. Severson, T.M. Markov, Y.-H. Liao, M.H. Chen, B. Cheong, N. Perkins, Z. Yang, et al., Closed-loop optimization of fast-charging protocols for batteries with machine learning, Nature 578 (7795) (2020) 397–402.

[28] Y. Che, Y. Zheng, Y. Wu, X. Sui, P. Bharadwaj, D.-I. Stroe, Y. Yang, X. Hu, R. Teodorescu, Data efficient health prognostic for batteries based on sequential information-driven probabilistic neural network, Appl. Energy 323 (2022) 119663.

[29] S. Hong, Y. Zeng, A health assessment framework of lithium-ion batteries for cyber defense, Appl. Soft Comput. 101 (2021) 107067.

[30] Z. Chen, Q. Xue, Y. Wu, S. Shen, Y. Zhang, J. Shen, Capacity prediction and validation of lithium-ion batteries based on long short-term memory recurrent neural network, IEEE Access 8 (2020) 172783–172798.

[31] F. Heinrich, M. Pruckner, Virtual experiments for battery state of health estimation based on neural networks and in-vehicle data, J. Energy Storage 48 (2022) 103856.

[32] F. Heinrich, P. Klapper, M. Pruckner, A comprehensive study on battery electric modeling approaches based on machine learning, Energy Inform. 4 (2021) 1–17.

[33] Y. Gong, X. Zhang, D. Gao, H. Li, L. Yan, J. Peng, Z. Huang, State-of-health estimation of lithium-ion batteries based on improved long short-term memory algorithm, J. Energy Storage 53 (2022) 105046.

[34] P. Ren, S. Wang, X. Chen, H. Zhou, C. Fernandez, D.-I. Stroe, A novel multiple training-scale dynamic adaptive cuckoo search optimized long short-term memory neural network and multi-dimensional health indicators acquisition strategy for whole life cycle health evaluation of lithium-ion batteries, Electrochim. Acta 435 (2022) 141404.

[35] J.C. Chen, T.-L. Chen, W.-J. Liu, C. Cheng, M.-G. Li, Combining empirical mode decomposition and deep recurrent neural networks for predictive maintenance of lithium-ion battery, Adv. Eng. Inform. 50 (2021) 101405.

[36] Z. Zhang, H. Min, H. Guo, Y. Yu, W. Sun, J. Jiang, H. Zhao, State of health estimation method for lithium-ion batteries using incremental capacity and long short-term memory network, J. Energy Storage 64 (2023) 107063.

[37] M. Lin, Y. You, W. Wang, J. Wu, Battery health prognosis with gated recurrent unit neural networks and hidden Markov model considering uncertainty quantification, Reliab. Eng. Syst. Saf. 230 (2023) 108978.

[38] N. Khan, F.U.M. Ullah, A. Ullah, M.Y. Lee, S.W. Baik, et al., Batteries state of health estimation via efficient neural networks with multiple channel charging profiles, IEEE Access 9 (2020) 7797–7813.

[39] Z. Chen, H. Zhao, Y. Zhang, S. Shen, J. Shen, Y. Liu, State of health estimation for lithium-ion batteries based on temperature prediction and gated recurrent unit neural network, J. Power Sources 521 (2022) 230892.

[40] K. Li, Y. Wang, Z. Chen, A comparative study of battery state-of-health estimation based on empirical mode decomposition and neural network, J. Energy Storage 54 (2022) 105333.

[41] Y. Tian, Q. Dong, J. Tian, X. Li, G. Li, K. Mehran, Capacity estimation of lithium-ion batteries based on optimized charging voltage section and virtual sample generation, Appl. Energy 332 (2023) 120516.

[42] S.W. Kim, K.-Y. Oh, S. Lee, Novel informed deep learning-based prognostics framework for on-board health monitoring of lithium-ion batteries, Appl. Energy 315 (2022) 119011.

[43] H.H. Goh, Z. Lan, D. Zhang, W. Dai, T.A. Kurniawan, K.C. Goh, Estimation of the state of health (SOH) of batteries using discrete curvature feature extraction, J. Energy Storage 50 (2022) 104646.

[44] B. Ma, H.-Q. Yu, W.-T. Wang, X.-B. Yang, L.-S. Zhang, H.-C. Xie, C. Zhang, S.-Y. Chen, X.-H. Liu, State of health and remaining useful life prediction for lithium-ion batteries based on differential thermal voltammetry and a long and short memory neural network, Rare Metals (2022) 1–17.

[45] B. Ma, S. Yang, L. Zhang, W. Wang, S. Chen, X. Yang, H. Xie, H. Yu, H. Wang, X. Liu, Remaining useful life and state of health prediction for lithium batteries based on differential thermal voltammetry and a deep-learning model, J. Power Sources 548 (2022) 232030.

[46] W. Li, N. Sengupta, P. Dechent, D. Howey, A. Annaswamy, D.U. Sauer, One-shot battery degradation trajectory prediction with deep learning, J. Power Sources 506 (2021) 230024.

[47] W. Li, N. Sengupta, P. Dechent, D. Howey, A. Annaswamy, D.U. Sauer, Online capacity estimation of lithium-ion batteries with deep long short-term memory networks, J. Power Sources 482 (2021) 228863.

[48] S. Zhao, C. Zhang, Y. Wang, Lithium-ion battery capacity and remaining useful life prediction using board learning system and long short-term memory neural network, J. Energy Storage 52 (2022) 104901.

[49] P. Li, Z. Zhang, Q. Xiong, B. Ding, J. Hou, D. Luo, Y. Rong, S. Li, 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 459 (2020) 228069.

[50] F.-K. Wang, Z.E. Amogne, J.-H. Chou, C. Tseng, Online remaining useful life prediction of lithium-ion batteries using bidirectional long short-term memory with attention mechanism, Energy 254 (2022) 124344.

[51] L. Zhang, T. Ji, S. Yu, G. Liu, Accurate prediction approach of SOH for lithium-ion batteries based on LSTM method, Batteries 9 (3) (2023) 177.

[52] W. Li, H. Zhang, B. van Vlijmen, P. Dechent, D.U. Sauer, Forecasting battery capacity and power degradation with multi-task learning, Energy Storage Mater. 53 (2022) 453–466.

[53] G. Cheng, X. Wang, Y. He, Remaining useful life and state of health prediction for lithium batteries based on empirical mode decomposition and a long and short memory neural network, Energy 232 (2021) 121022.

[54] Z. Chen, L. Chen, W. Shen, K. Xu, Remaining useful life prediction of lithium-ion battery via a sequence decomposition and deep learning integrated approach, IEEE Trans. Veh. Technol. 71 (2) (2021) 1466–1479.

[55] W. Zhang, X. Li, X. Li, Deep learning-based prognostic approach for lithium-ion batteries with adaptive time-series prediction and on-line validation, Measurement 164 (2020) 108052.

[56] J. Hong, Z. Wang, W. Chen, L. Wang, P. Lin, C. Qu, Online accurate state of health estimation for battery systems on real-world electric vehicles with variable driving conditions considered, J. Clean. Prod. 294 (2021) 125814.

[57] N. Yang, Z. Song, H. Hofmann, J. Sun, Robust state of health estimation of lithium-ion batteries using convolutional neural network and random forest, J. Energy Storage 48 (2022) 103857.

[58] H. Ruan, J. Chen, W. Ai, B. Wu, Generalised diagnostic framework for rapid battery degradation quantification with deep learning, Energy AI 9 (2022) 100158.

[59] S. Shen, M. Sadoughi, X. Chen, M. Hong, C. Hu, A deep learning method for online capacity estimation of lithium-ion batteries, J. Energy Storage 25 (2019) 100817.

[60] R. Xiong, J. Tian, W. Shen, J. Lu, F. Sun, Semi-supervised estimation of capacity degradation for lithium ion batteries with electrochemical impedance spectroscopy, J. Energy Chem. 76 (2023) 404–413.

[61] T. Pradyumna, K. Cho, M. Kim, W. Choi, Capacity estimation of lithium-ion batteries using convolutional neural network and impedance spectra, J. Power Electron. 22 (5) (2022) 850–858.

[62] N. Costa, L. Sanchez, D. Ansean, M. Dubarry, Li-ion battery degradation modes diagnosis via convolutional neural networks, J. Energy Storage 55 (2022) 105558.

[63] S. Sohn, H.-E. Byun, J.H. Lee, Two-stage deep learning for online prediction of knee-point in Li-ion battery capacity degradation, Appl. Energy 328 (2022) 120204.

[64] S. Ji, J. Zhu, Z. Lyu, H. You, Y. Zhou, L. Gu, J. Qu, Z. Xia, Z. Zhang, H. Dai, Deep learning enhanced lithium-ion battery nonlinear fading prognosis, J. Energy Chem. (2023).

[65] S. Park, H. Lee, Z.K. Scott-Nevros, D. Lim, D.-H. Seo, Y. Choi, H. Lim, D. Kim, Deep-learning based spatio-temporal generative model on assessing state-of-health for Li-ion batteries with partially-cycled profiles, Mater. Horiz. (2023).

[66] J. Yao, T. Han, Data-driven lithium-ion batteries capacity estimation based on deep transfer learning using partial segment of charging/discharging data, Energy (2023) 127033.

[67] G. Fan, X. Zhang, Battery capacity estimation using 10-second relaxation voltage and a convolutional neural network, Appl. Energy 330 (2023) 120308.

[68] Z. Deng, X. Hu, Y. Xie, L. Xu, P. Li, X. Lin, X. Bian, Battery health evaluation using a short random segment of constant current charging, Iscience 25 (5) (2022) 104260.

[69] J. Tian, R. Xiong, W. Shen, J. Lu, X.-G. Yang, Deep neural network battery charging curve prediction using 30 points collected in 10 min, Joule 5 (6) (2021) 1521–1534.

[70] J. Tian, R. Xiong, W. Shen, J. Lu, F. Sun, Flexible battery state of health and state of charge estimation using partial charging data and deep learning, Energy Storage Mater. 51 (2022) 372–381.

[71] S. Saxena, L. Ward, J. Kubal, W. Lu, S. Babinec, N. Paulson, A convolutional neural network model for battery capacity fade curve prediction using early life data, J. Power Sources 542 (2022) 231736.

[72] D. Zhou, B. Wang, Battery health prognosis using improved temporal convolutional network modeling, J. Energy Storage 51 (2022) 104480.

[73] S. Bockrath, V. Lorentz, M. Pruckner, State of health estimation of lithium-ion batteries with a temporal convolutional neural network using partial load profiles, Appl. Energy 329 (2023) 120307.

[74] H. Wang, J. Li, X. Liu, J. Rao, Y. Fan, X. Tan, Online state of health estimation for lithium-ion batteries based on a dual self-attention multivariate time series prediction network, Energy Rep. 8 (2022) 8953–8964.

[75] C. Qian, B. Xu, Q. Xia, Y. Ren, B. Sun, Z. Wang, SOH prediction for lithium-ion batteries by using historical state and future load information with an AM-seq2seq model, Appl. Energy 336 (2023) 120793.

[76] F. Wang, Z. Zhao, J. Ren, Z. Zhai, S. Wang, X. Chen, A transferable lithium-ion battery remaining useful life prediction method from cycle-consistency of degradation trend, J. Power Sources 521 (2022) 230975.

[77] F. Wang, Z. Zhao, Z. Zhai, S. Wang, B. Ding, X. Chen, Remaining useful life prediction of lithium-ion battery based on cycle-consistency learning, in: 2021 International Conference on Sensing, Measurement & Data Analytics in the Era of Artificial Intelligence, ICSMD, IEEE, 2021, pp. 1–6.

[78] S. Kim, Y.Y. Choi, J.-I. Choi, Impedance-based capacity estimation for lithium-ion batteries using generative adversarial network, Appl. Energy 308 (2022) 118317.

[79] Z. Cui, X. Gao, J. Mao, C. Wang, Remaining capacity prediction of lithium-ion battery based on the feature transformation process neural network, Expert Syst. Appl. 190 (2022) 116075.

[80] M.B. Mckay, B. Wetton, R.B. Gopaluni, Learning physics based models of lithium-ion batteries, IFAC-PapersOnLine 54 (3) (2021) 97–102.

[81] Q. Gong, P. Wang, Z. Cheng, An encoder-decoder model based on deep learning for state of health estimation of lithium-ion battery, J. Energy Storage 46 (2022) 103804.

[82] S. Li, H. He, C. Su, P. Zhao, Data driven battery modeling and management method with aging phenomenon considered, Appl. Energy 275 (2020) 115340.

[83] S. Zhang, B. Zhai, X. Guo, K. Wang, N. Peng, X. Zhang, 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 26 (2019) 100951.

[84] Z. Wei, X. Han, J. Li, State of health assessment for echelon utilization batteries based on deep neural network learning with error correction, J. Energy Storage 51 (2022) 104428.

[85] M. Cao, T. Zhang, J. Wang, Y. Liu, A deep belief network approach to remaining capacity estimation for lithium-ion batteries based on charging process features, J. Energy Storage (Apr.) (2022) 48.

[86] Y. Wang, K. Li, P. Peng, Z. Chen, Health diagnosis for lithium-ion battery by combining partial incremental capacity and deep belief network during insufficient discharge profile, IEEE Trans. Ind. Electron. (2022).

[87] M. Lin, X. Zeng, J. Wu, State of health estimation of lithium-ion battery based on an adaptive tunable hybrid radial basis function network, J. Power Sources 504 (2021) 230063.

[88] T. Xu, Z. Peng, L. Wu, A novel data-driven method for predicting the circulating capacity of lithium-ion battery under random variable current, Energy 218 (2021) 119530.

[89] J. Jia, S. Yuan, Y. Shi, J. Wen, X. Pang, J. Zeng, Improved sparrow search algorithm optimization deep extreme learning machine for lithium-ion battery state-of-health prediction, Iscience 25 (4) (2022) 103988.

[90] A. Xue, W. Yang, X. Yuan, B. Yu, C. Pan, Estimating state of health of lithium-ion batteries based on generalized regression neural network and quantum genetic algorithm, Appl. Soft Comput. 130 (2022) 109688.

[91] S. Mian Qaisar, A.E.E. AbdelGawad, K. Srinivasan, Event-driven acquisition and machine-learning-based efficient prediction of the Li-ion battery capacity, SN Comput. Sci. 3 (2022) 1–6.

[92] L. Song, K. Zhang, T. Liang, X. Han, Y. Zhang, Intelligent state of health estimation for lithium-ion battery pack based on big data analysis, J. Energy Storage 32 (2020) 101836.

[93] S. Maleki, A. Mahmoudi, A. Yazdani, Knowledge transfer-oriented deep neural network framework for estimation and forecasting the state of health of the lithium-ion batteries, J. Energy Storage 53 (2022) 105183.

[94] Q. Lin, H. Li, Q. Chai, F. Cai, Y. Zhan, Simultaneous and rapid estimation of state of health and state of charge for lithium-ion battery based on response characteristics of load surges, J. Energy Storage 55 (2022) 105495.

[95] S. Pepe, J. Liu, E. Quattrocchi, F. Ciucci, Neural ordinary differential equations and recurrent neural networks for predicting the state of health of batteries, J. Energy Storage 50 (2022) 104209.

[96] H. Zhang, W. Tang, W. Na, P.-Y. Lee, J. Kim, Implementation of generative adversarial network-CLS combined with bidirectional long short-term memory for lithium-ion battery state prediction, J. Energy Storage 31 (2020) 101489.

[97] X.-Y. Yao, G. Chen, L. Hu, M. Pecht, A multi-model feature fusion model for lithium-ion battery state of health prediction, J. Energy Storage 56 (2022) 106051.

[98] Y. Che, Z. Deng, P. Li, X. Tang, K. Khosravinia, X. Lin, X. Hu, State of health prognostics for series battery packs: A universal deep learning method, Energy 238 (2022) 121857.

[99] Y. Toughzaoui, S.B. Toosi, H. Chaoui, H. Louahlia, R. Petrone, S. Le Masson, H. Gualous, State of health estimation and remaining useful life assessment of lithium-ion batteries: A comparative study, J. Energy Storage 51 (2022) 104520.

[100] C. Wang, Y. Ding, N. Yan, L. Ma, J. Ma, C. Lu, C. Yang, Y. Su, J. Chong, H. Jin, et al., A novel long-term degradation trends predicting method for multi-formulation Li-ion batteries based on deep reinforcement learning, Adv. Eng. Inform. 53 (2022) 101665.

[101] A. Kara, A data-driven approach based on deep neural networks for lithium-ion battery prognostics, Neural Comput. Appl. 33 (20) (2021) 13525–13538.

[102] Z. Zhu, Q. Yang, X. Liu, D. Gao, Attention-based CNN-bilstm for SOH and RUL estimation of lithium-ion batteries, J. Algorithms Comput. Technol. 16 (2022) 17483026221130598.

[103] J. He, Y. Tian, L. Wu, A hybrid data-driven method for rapid prediction of lithium-ion battery capacity, Reliab. Eng. Syst. Saf. 226 (2022) 108674.

[104] Y. Wei, D. Wu, Prediction of state of health and remaining useful life of lithium-ion battery using graph convolutional network with dual attention mechanisms, Reliab. Eng. Syst. Saf. 230 (2023) 108947.

[105] Y. Jiang, Y. Chen, F. Yang, W. Peng, State of health estimation of lithium-ion battery with automatic feature extraction and self-attention learning mechanism, J. Power Sources 556 (2023) 232466.

[106] Y. Wei, D. Wu, Prediction of state of health and remaining useful life of lithium-ion battery using graph convolutional network with dual attention mechanisms, Reliab. Eng. Syst. Saf. 230 (2023) 108947.

[107] X. Gu, K. See, P. Li, K. Shan, Y. Wang, L. Zhao, K.C. Lim, N. Zhang, A novel state-of-health estimation for the lithium-ion battery using a convolutional neural network and transformer model, Energy 262 (2023) 125501.

[108] P. Shrivastava, T.K. Soon, M.Y.I.B. Idris, S. Mekhilef, Overview of model-based online state-of-charge estimation using Kalman filter family for lithium-ion batteries, Renew. Sustain. Energy Rev. 113 (2019) 109233.

[109] M.A. Hannan, M.H. Lipu, A. Hussain, A. Mohamed, A review of lithium-ion battery state of charge estimation and management system in electric vehicle applications: Challenges and recommendations, Renew. Sustain. Energy Rev. 78 (2017) 834–854.

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

[111] M. Berecibar, I. Gandiaga, I. Villarreal, N. Omar, J. Van Mierlo, P. Van den Bossche, Critical review of state of health estimation methods of Li-ion batteries for real applications, Renew. Sustain. Energy Rev. 56 (2016) 572–587.

[112] Q. Lin, J. Wang, R. Xiong, W. Shen, H. He, Towards a smarter battery management system: A critical review on optimal charging methods of lithium ion batteries, Energy 183 (2019) 220–234.

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

[114] S. Yang, C. Zhang, J. Jiang, W. Zhang, L. Zhang, Y. Wang, Review on state-of-health of lithium-ion batteries: Characterizations, estimations and applications, J. Clean. Prod. 314 (2021) 128015.

[115] M.-F. Ge, Y. Liu, X. Jiang, J. Liu, A review on state of health estimations and remaining useful life prognostics of lithium-ion batteries, Measurement 174 (2021) 109057.

[116] J. Zhao, Y. Zhu, B. Zhang, M. Liu, J. Wang, C. Liu, X. Hao, Review of state estimation and remaining useful life prediction methods for lithium–Ion batteries, Sustainability 15 (6) (2023) 5014.

[117] Y. Che, X. Hu, X. Lin, J. Guo, R. Teodorescu, Health prognostics for lithium-ion batteries: mechanisms, methods, and prospects, Energy Environ. Sci. (2023).

[118] M.H. Lipu, S. Ansari, M.S. Miah, S.T. Meraj, K. Hasan, A. Shihavuddin, M. Hannan, K.M. Muttaqi, A. Hussain, Deep learning enabled state of charge, state of health and remaining useful life estimation for smart battery management system: Methods, implementations, issues and prospects, J. Energy Storage 55 (2022) 105752.

[119] C. Lin, J. Xu, X. Mei, Improving state-of-health estimation for lithium-ion batteries via unlabeled charging data, Energy Storage Mater. 54 (2023) 85–97.

[120] B. Saha, K. Goebel, Battery Data Set, NASA Ames Research Center, Moffett Field, CA, 2007, NASA Ames Prognostics Data Repository, https://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/#battery.

[121] K.A. Severson, P.M. Attia, N. Jin, N. Perkins, B. Jiang, Z. Yang, M.H. Chen, M. Aykol, P.K. Herring, D. Fraggedakis, et al., Data-driven prediction of battery cycle life before capacity degradation, Nature Energy 4 (5) (2019) 383–391.

[122] C. Birkl, Diagnosis and Prognosis of Degradation in Lithium-Ion Batteries (Ph.D. thesis), University of Oxford, 2017.

[123] W. He, N. Williard, M. Osterman, M. Pecht, Prognostics of lithium-ion batteries based on Dempster–Shafer theory and the Bayesian Monte Carlo method, J. Power Sources 196 (23) (2011) 10314–10321.

[124] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning Internal Representations by Error Propagation, Tech. Rep., California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[125] K. Fukushima, Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position, Biol. Cybern. 36 (4) (1980) 193–202.

[126] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.

[127] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[128] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.

[129] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, 2014, arXiv preprint arXiv:1406.1078.

[130] V. Mnih, N. Heess, A. Graves, et al., Recurrent models of visual attention, Adv. Neural Inf. Process. Syst. 27 (2014).

[131] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, 2014, arXiv preprint arXiv:1409.0473.

[132] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).

[133] D. Zhao, Z. Zhou, S. Tang, Y. Cao, J. Wang, P. Zhang, Y. Zhang, Online estimation of satellite lithium-ion battery capacity based on approximate belief rule base and hidden Markov model, Energy 256 (2022) 124632.

[134] C. Lin, J. Xu, X. Mei, Improving state-of-health estimation for lithium-ion batteries via unlabeled charging data, Energy Storage Mater. 54 (2023) 85–97.

[135] C.B. Salucci, A. Bakdi, I.K. Glad, E. Vanem, R. De Bin, A novel semi-supervised learning approach for state of health monitoring of maritime lithium-ion batteries, J. Power Sources 556 (2023) 232429.

[136] Z. Zhao, Q. Zhang, X. Yu, C. Sun, S. Wang, R. Yan, X. Chen, Applications of unsupervised deep transfer learning to intelligent fault diagnosis: A survey and comparative study, IEEE Trans. Instrum. Meas. 70 (2021) 1–28.

[137] F. Wang, Z. Zhao, Z. Zhai, Y. Guo, H. Xi, S. Wang, X. Chen, Feature disentanglement and tendency retainment with domain adaptation for lithium-ion battery capacity estimation, Reliab. Eng. Syst. Saf. 230 (2023) 108897.

[138] K.Q. Zhou, Y. Qin, C. Yuen, Transfer-learning-based state-of-health estimation for lithium-ion battery with cycle synchronization, IEEE/ASME Trans. Mechatronics (2022).

[139] T. Han, Z. Wang, H. Meng, End-to-end capacity estimation of lithium-ion batteries with an enhanced long short-term memory network considering domain adaptation, J. Power Sources 520 (2022) 230823.

[140] Z. Ye, J. Yu, State-of-health estimation for lithium-ion batteries using domain adversarial transfer learning, IEEE Trans. Power Electron. 37 (3) (2021) 3528–3543.

[141] Z. Zhi, W. Fujin, D. Yi, M. Peiyu, Z. Zhibin, C. Xuefeng, Hierarchical alignment transfer learning for lithium-ion battery capacity estimation, Energy Storage Sci. Technol. 1.

[142] H. Zhang, Y. Su, F. Altaf, T. Wik, S. Gros, Interpretable battery cycle life range prediction using early degradation data at cell level, 2022, arXiv preprint arXiv:2204.12420.

[143] G. Lee, J. Kim, C. Lee, State-of-health estimation of li-ion batteries in the early phases of qualification tests: An interpretable machine learning approach, Expert Syst. Appl. 197 (2022) 116817.

[144] G. Lee, D. Kwon, C. Lee, A convolutional neural network model for SOH estimation of Li-ion batteries with physical interpretability, Mech. Syst. Signal Process. 188 (2023) 110004.

[145] F. Wang, Z. Zhao, Z. Zhai, Z. Shang, R. Yan, X. Chen, Explainability-driven model improvement for SOH estimation of lithium-ion battery, Reliab. Eng. Syst. Saf. 232 (2023) 109046.

[146] R.G. Nascimento, M. Corbetta, C.S. Kulkarni, F.A. Viana, Hybrid physics-informed neural networks for lithium-ion battery modeling and prognosis, J. Power Sources 513 (2021) 230526.

[147] A. Thelen, Y.H. Lui, S. Shen, S. Laflamme, S. Hu, H. Ye, C. Hu, Integrating physics-based modeling and machine learning for degradation diagnostics of lithium-ion batteries, Energy Storage Mater. 50 (2022) 668–695.

[148] J. Shi, A. Rivera, D. Wu, Battery health management using physics-informed machine learning: Online degradation modeling and remaining useful life prediction, Mech. Syst. Signal Process. 179 (2022) 109347.

[149] H. Tu, S. Moura, Y. Wang, H. Fang, Integrating physics-based modeling with machine learning for lithium-ion batteries, Appl. Energy 329 (2023) 120289.

[150] M. Aykol, C.B. Gopal, A. Anapolsky, P.K. Herring, B. van Vlijmen, M.D. Berliner, M.Z. Bazant, R.D. Braatz, W.C. Chueh, B.D. Storey, Perspective—combining physics and machine learning to predict battery lifetime, J. Electrochem. Soc. 168 (3) (2021) 030525.