Article:
Zhao, Y., Deng, J., Liu, P. et al. (5 more authors) (2025) Enhancing battery durable operation: Multi-fault diagnosis and safety evaluation in series-connected lithium-ion battery systems. Applied Energy, 377 (Part C). 124632. ISSN 0306-2619

https://doi.org/10.1016/j.apenergy.2024.124632

Enhancing Battery Durable Operation: Multi-Fault Diagnosis and Safety Evaluation in Series-Connected Lithium-ion Battery Systems

Yiwen Zhao[a,b], Junjun Deng[a,b], Peng Liu[a,b,c], Lei Zhang[a,b], Dingsong Cui[d], Qiushi Wang[a,b,e], Zhenyu Sun[f,g,*] and Zhenpo Wang[a,b,c,]**

a. National Engineering Research Center of Electric Vehicles, Beijing Institute of Technology, Beijing, 100081, China
b. Beijing Co-innovation Centre for Electric Vehicles, Beijing, 100081, China
c. Beijing Institute of Technology Chongqing Innovation Center, Chongqing, 401120, China
d. Institute for Transport Studies, University of Leeds, LS2 9JT, United Kingdom
e. Beijing Automotive Research Institute Co., Ltd, Beijing, 100176, China
f. South China University of Technology, Guangzhou, 510640, China
g. Sunwoda Power Technology Co., Ltd., Shenzhen, 518107, China

Corresponding Author:

**Zhenpo Wang (wangzhenpo@bit.edu.cn) and *Zhenyu Sun (bitzhenyu@163.com)

Corresponding Address:

National Engineering Research Center of Electric Vehicles, Beijing Institute of Technology, Beijing, 100081, China

Abstract

Precise fault identification and evaluation of battery systems are indispensably required to facilitate safe and durable operation for electric vehicles. With the core objective of addressing the challenges of inaccurate evaluation and misdiagnoses of multi-fault in existing methods, this paper proposes a deep-learning-powered diagnosis and evaluation scheme for series-connected battery systems. First, we conduct series-connected cycling experiments to simulate the two most common faults including capacity anomaly fault and short circuit fault happening concurrently to observe the failure phenomena of different faulty batteries and fault-free batteries. Then, the evolutional processes of various faults are analyzed and compared for a deeper understanding of the battery fault mechanism. In addition, we establish an elaborate deep-learning-based model, achieving satisfactory realizations on predicting the reference voltage (with the mean square error of 7.84×10^{-5} V) while categorizing the current fault state (with an accuracy of 98.2%). At last, a comprehensive fault identification and quantification strategy is constructed to minimize the misdiagnosis. All proposed methodologies demonstrate the advancement compared to other state-of-the-art algorithms. The results are thoroughly validated with two different experimental datasets and real-world cloud vehicle datasets, affirming the efficiency and practical applicability, contributing to enhancing the active safety capabilities of battery systems.

Keywords: Lithium-ion Batteries; Multi-Fault Diagnosis; Deep-Learning Technologies; Safety Evaluation Strategy

1 Introduction

The world is becoming increasingly electrified [1–3]. Transportation, renewable energy storage systems and mobile devices, especially for ramping electric vehicle (EV) deployment, are calling for much better batteries [4,5]. The commercialization of lithium-ion batteries (LIBs) has accelerated the electrification process of vehicles [6–8]. To develop a longer all-electric driving range, series/parallel assembly with higher battery power density (250-693 Wh/L and 100-265 Wh/kg for LIBs [9]) individual cells have highlighted an increasing and continuing concern on safety issues [10]. The risk of hazardous failure of LIBs from reputable manufacturers is less portable but needs to be understood solidly for regulators [11]. Accurate and advanced pre- and diagnosing potential faults is extremely challenging prior to a destructive explosion of safety issues. Additionally, since the LIB is essentially an electrochemical energy storage system, the safety issues of LIBs are directly related to the materials of key internal components and their chemical stability. This will ultimately be reflected in abnormal variations of external parameters. Therefore, we focus on resolving the diagnosis challenges by dealing with external characteristics of LIBs in this paper.

The most catastrophic failure mode of LIBs is thermal runaway (TR) [12], which has a high probability of evolving gradually from the inconsistencies of the battery system in realistic operation [13,14]. This condition can be caused and enlarged by continuous overcharge/overdischarge [15,16], short circuit (SC) [17], connection issues, sensor fault [18], capacity anomaly (CA) [19], etc., leading to rapid and uncontrolled increase in temperature, and eventually develops into TR [20]. An SC in a LIB occurs when there is an unintended connection between the positive and negative terminals, typically caused by manufacturing defects, physical damage and so on [21]. Among all the known types of battery failure modes, the SC tops the list of the major safety concerns for LIBs [22]. A micro SC fault only manifests negligible abnormalities in the early stage, while it will be a severe deterioration after a long

evolution process [23], causing improving self-discharge rate and calorific value. Similarly, the CA fault will directly cause inconsistency of the battery system, while the cells in the battery pack will experience differing rates of degradation, as illustrated in our previous work [24]. Frequent failure abuse for cells as well as unbalanced initial cell capacity in the battery pack can result in the CA fault. Nevertheless, motivated by the confusing external properties and similar evolutionary progress of these faults, this paper aims to enhance the safety risk early-warning capability of battery systems considering these two electrical abuse conditions deeply.

An associated strand of research has diagnosed the faults with model- and data-driven-based methods. The model-based methods focus on employing battery models to generate residuals, which help in identifying the presence of faults [25,26]. These methods are widely appreciated for their high precision in detecting faults. However, the complexity involved in developing and solving these models cannot be overlooked. The mathematical models require a deep understanding of the battery's internal mechanisms and intricate computations, making them resource-intensive and challenging to implement on a large scale. Better yet, data-driven-based methods straightly cope with the battery running data, eliminating the need for constructing explicit mathematical models or possessing in-depth knowledge of the battery's internal dynamics [24,27,28]. These methods leverage advanced algorithms and statistical tools to diagnose faults [29]. Specifically, deep learning (DL) algorithms, traditional machine learning algorithms, and various statistical instruments are effectively utilized for this purpose [30–32]. The reliability of these methods heavily depends on the quality of the data being analyzed. Nevertheless, advanced data collection technology for batteries enables it possible for a widespread data-driven application.

Focusing on the model application scenario, it can be noted that plentiful studies have only investigated the specific battery fault diagnosis, such as single SC fault diagnosis [33,34].

However, various types of battery faults generally coincide in real-world applications due to uncertain external environments and operating situations of EVs. These faults are generally insidious and similar, bringing challenges to differentiate and detect promptly. Thus, diagnosis methods for a single fault are not universal enough, while multi-fault diagnosis will be difficult to implement. That is the main gap that we find in previous studies and the first issue that we aim to solve in this paper. Moreover, a battery management system (BMS) can only detect obvious faults by thresholds such as drastic over/under voltage, overcurrent and overtemperature. The external characteristics of most of the micro failures in the early stage are tiny and hidden due to a long-term evolutionary process before the TR of the battery, which cannot be pre-detected by the BMS in time. That is the second problem covered and investigated in this paper. Thus, a comprehensive multi-fault diagnosis scheme based on the fault mechanism and DL-powered technologies is proposed to solve the above problems. The main contributions of the paper are listed as follows:

a. Cycles of multi-fault simulation experiments are conducted to concurrently observe the characteristics and discrepancies of cells, enabling us to analyze the evolution mechanism of different faults, providing a solid fault-mechanism-based foundation and labeled insights for constructing a reasonable diagnostic scheme.

b. The entire diagnostic scheme only adopts directly measurable parameters and as few but sufficiently effective fault indicators as possible for feature extraction and model building with no need for redundant measurement sensors. The simplicity and practical application possibilities can be fully guaranteed.

c. Taking into account both the abnormal changes at the individual cell level and the potential inconsistencies occurring within cells at the module level, we have separately constructed a highly accurate Predictor and a well-performed Classifier with the assistance

of DL algorithms. The proposed models are capable of providing complete diagnostics for the entire battery system.

d. Our paper further extends the application of time windows and statistical methods to determine the type of the failure, especially focusing on quantifying the degree of multiple faults. The constructed safety evaluation strategy can effectively avoid the probability of misdiagnosis. The credibility and practical implementation of the proposed models can also be improved efficiently.

e. All technical processes of the proposed diagnostic framework are validated in the real-world operational vehicle dataset and perform well, demonstrating strong applicability.

The remainder of the paper is structured as demonstrated: **Section 2 (Framework Overview)** provides an overview of the motivation and the technological framework of the article. **Section 3 (Experimental and cloud data)** introduces the procedures of fault simulation experiments and cloud EV operational data. **Section 4 (Methods)** details the construction of the DL-powered fault identification and quantification strategy. **Section 5** (**Results and Discussions**) evidences the diagnostic results and applicability of the proposed framework include while dialectically outlining the superiorities of the proposed methodology. Finally, **Section 6** (**Conclusions**) summarizes the overall technological process. The rest of the necessary materials are placed in the **Supplementary**.

2 Framework Overview

The primary motivation for this work is to provide as comprehensive and effective a diagnosis approach of the battery as possible before it causes further harm and to make it practical application. Dedicated to diagnosing multi- fault in battery systems, we carry out three main efforts as outlined in **Fig.1**: **(a) Experimental and cloud data:** In order to observe the behavior of simultaneous faults in a series-connected battery system and to furnish theoretical and phenomenological insights for the follow-up fault diagnosis, we conduct cyclic multi-fault tests and make further fault evolution mechanism analysis for a reasonable fault feature selection. The cloud EV operational data are briefly introduced for the framework validation. **(b) DL model construction：** Leveraging the power of DL algorithms and the accessibility of directly measurable parameters, a Predictor and a Classifier are built for realizing the prediction of reference voltages at the cell level and the classification of various faults at the module level, respectively. All proposed models have been rigorously trained, tested, validated and compared to current state-of-the-art algorithms. **(c) Safety evaluation scheme:** Considering maximizing the practical applicability the overall diagnostic framework, we contribute to developing a rule for identifying faults to minimize misdiagnosis while at the same time quantifying and ranking the fault risk.

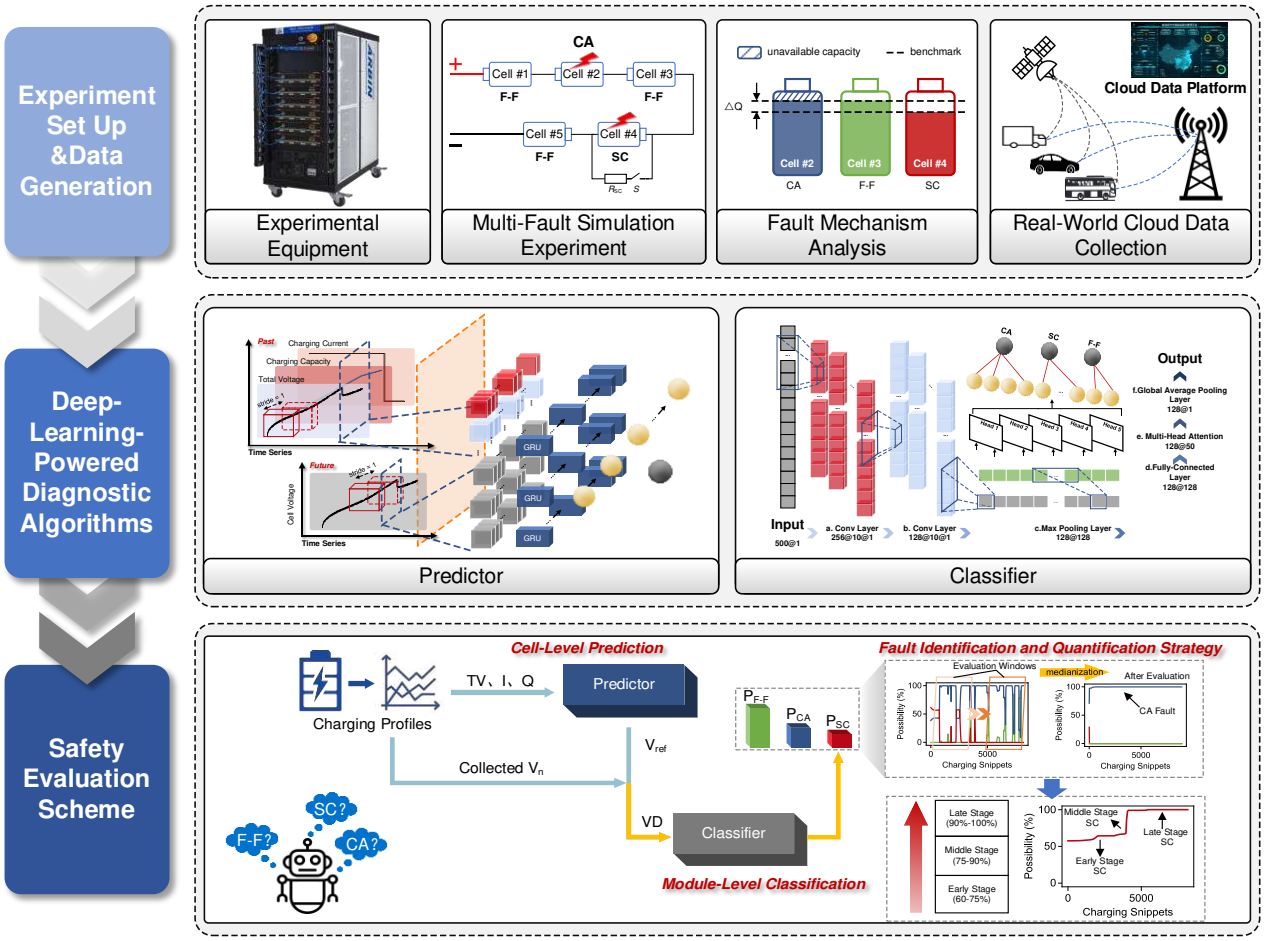

Fig.1 Overall workflow of the proposed multi-fault diagnosis framework.

3 Experimental and Cloud Data

3.1 Experiment Procedure and Data Generation

This paper delves into the examination of commercially available cylindrical 18650 NCR/graphite LIBs, the specifications of the batteries are illustrated in **Table 1**. A five-cycle series-connected test including multiple faults injection is designed to generate the basic data for this work. The detailed battery charge/discharge strategies are provided in **Supplementary Note 1** and **Fig.S1**. The experimental environment and example data of load and response are displayed in **Figs.2 a-c**, respectively. Specifically, the concrete simulation circuit is designed as shown in **Fig.2 d** that three F-F cells without any fault injection (#1, #3 and #5), one cell with a lower initial capacity simulating a CA fault (#2) and one cell paralleled with an external resistor to replicate an SC fault (#4) are involved in the experiment. An SC

of the battery can be regarded as being connected in parallel with a small resistance load, which will result in an instantaneous overcurrent and voltage reduction of the battery itself. Therefore, we choose to rationally simulate the triggering of SC by connecting resistors in parallel.

Table. 1 Specifications of the Cylindrical Ternary 18650 Batteries in the Experiments.

Items	Specifications	Notes
Manufacturer Code	NCR18650BD	Manufactured by Panasonic
Cathode	$Li(NiCoAl)O_2$	-
Anode	Graphite	-
Discharge Cut-off Voltage	2.5V	-
Charge Cut-off Voltage	4.2V	-
Nominal Voltage	3.6V	0.61A discharge at 25℃
Nominal Rated Capacity	3000mAh	0.61A discharge at 20℃
Max Continuous Discharge Current	10A	0~+40℃
Max Charge Current	1.5A	-
Weight	less than 49g	-
Internal Resistance	less than 35mΩ	AC impedance 1kHz

Specifically, the initial capacity of Cell #2 is intentionally reduced to nearly 95% of the other freshly connected cells in the fault-simulation circuit. R_{SC} is set at 50Ω, representing the external resistance of Cell #4. It is activated during the initial charging snippets, approximately 1,200 seconds into the test. It's worth mentioning that an additional testing dataset with different parameters is conducted for further validation. These new tests closely follow the experimental procedure described above**.** However, the state of health (SOH) of Cell #2 is decreased to 92%, while the external resistance of Cell #4 is increased to 100Ω. The detailed information of two groups of tests is displayed in **Table. 2**.

Table. 2 Basic Cell Information of the Cyclic Series-Connected Test

Group	Cell No.	Initial Capacity (mAh)	Fault Injection Type	Time of Fault Injection (sec)
A	#1	2928	F-F	-
	#2	**2830**	**CA Fault (95%SOH)**	-
	#3	2925	F-F	-
	#4	2936	**SC Fault (50Ω)**	1,200 (±0.05sec)
	#5	2932	F-F	-
B	#1	2928	F-F	-
	#2	**2760**	**CA Fault (92%SOH)**	-
	#3	2925	F-F	-
	#4	2936	**SC Fault (100Ω)**	1,200 (±0.05sec)
	#5	2932	F-F	-

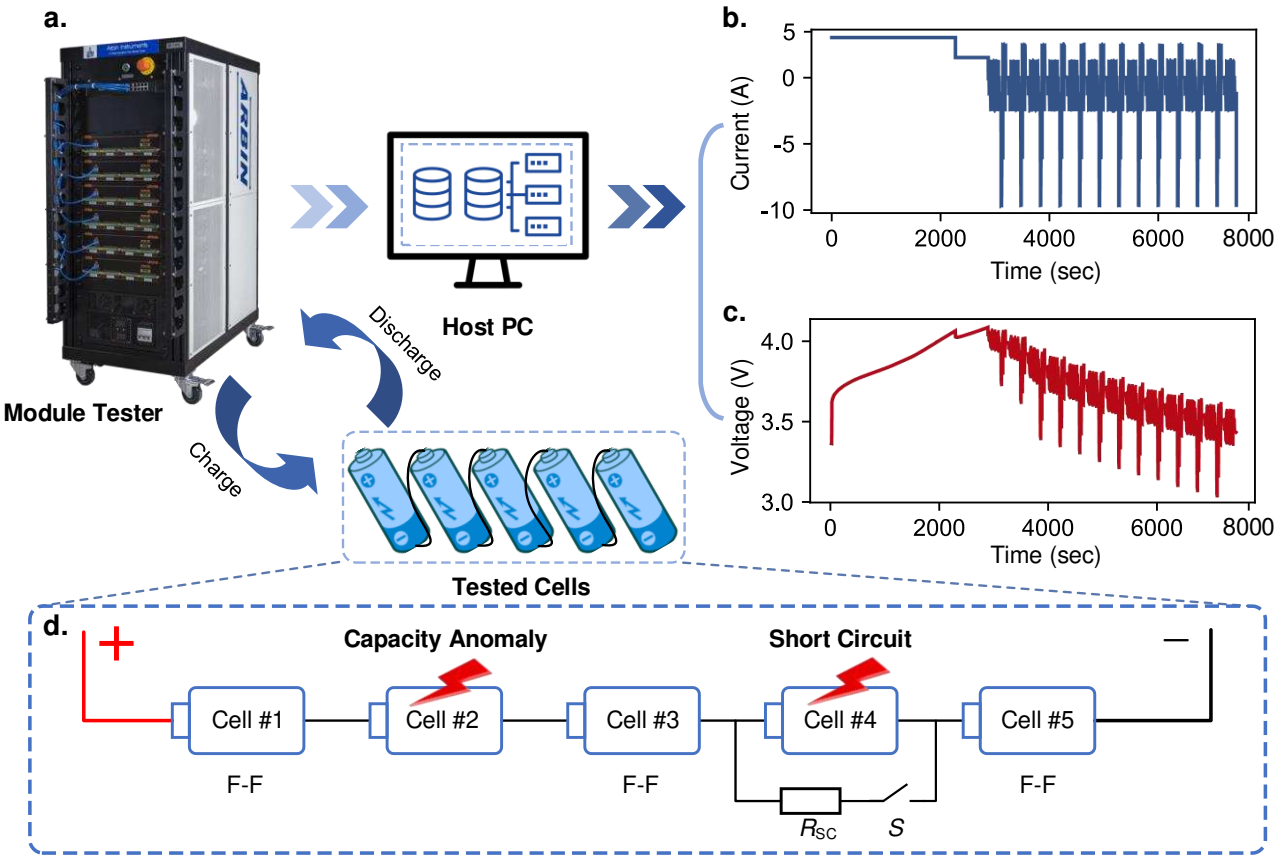

Fig.2 The schematic illustration of the experimental steps. a The experimental environment established by module tester (used for charge and discharge the batteries), host PC (data collection) and batteries, 5 cycles are conducted for observation; Detailed **b** load profiles and **c** voltage responses of the tested cells. **d** Specific fault simulation circuit with 5 tested battery cells (18650 NCR/graphite LIBs), assembled with 3 F-F battery cells (#1, #3 and #5), 1 CA battery cell (#2) and 1 SC battery cell (#4).

3.2 Understanding the Fault Evolution Mechanism

Our primary focus revolves around diagnosing micro electrical faults commonly encountered in battery systems, specifically SC and CA faults. As our cyclic experiment progresses, cells afflicted with various fault types exhibit distinct deterioration mechanisms. **Fig.3 a.** illustrates the voltage responses during a multi-stage constant current (CC) experiment within a sample series-connected battery, comprising an SC cell, a CA cell and an F-F cell. Zooming in on the inception of SC triggering (marked by the activation of switch S) in **Fig.3 b**, it can be observed a voltage drop resulting from current leakage, measuring approximately 2×10^{-3} V. This micro drop can be attributed to the application of an external resistance of a negligible magnitude. **Fig.3 c** presents a benchmark scenario involving the F-F cell during the charging phase, highlighting the occurrence of cell-to-cell inconsistencies. The voltage difference (*VD)* between the CA cell and the F-F cell (as well as between the SC cell and the F-F cell) is measured to quantify this variation. Formulation (1) provides a case of Cell #3 to calculate the *VD*:

$$VD_n(t) = \left|V_{ref}(t) - V_n(t)\right| \tag{1}$$

wherein, V_{ref} demonstrates the reference voltage of the battery system, which is in this case calculated from an F-F cell for standard; V_n is the *n*-th cell needed being detected; VD_n is the *VD* of the *n*-th cell; *t* refers to the sample time.

Furthermore, as depicted in **Fig.3 d**, we aim to elucidate the differentiation observed in cell behavior concerning capacity changes during charging under different failure modes. Typically, cells arranged in a battery series commence their charge-discharge cycle with an identical initial state of charge (SOC). Owing to the inherent characteristics of a series-connected configuration, both the CA cell and the F-F cell are charged to the same capacity, albeit exhibiting divergent voltage profiles. Notably, the voltage of CA cells experiences a

more rapid ascent during the charging process due to the less initial capacity. Concurrently, the SC cell exhibits a reduced charging capacity in comparison to the F-F cell, primarily attributable to the fact that SC leads to current leakage. Consequently, a battery afflicted with an SC fault during the charging process typically manifests a lower voltage level.

As mentioned above in **Fig.3 c**, *VDs* respond sensitively to injected faults of the experiment in this paper. Thus, the faulty progress by inter-cell *VDs* is further analyzed in **Fig.4.** We count the *VDs* of cells with different failure modes cycle by cycle to illustrate and compare the evolutionary process of faults involved in this paper. During the experiment, the CA and F-F cell show a relatively stable distribution, while the CA cell shows a wider range than that of the F-F cell. Differently, as the cycle deepens, the SC performs an incrementally larger, while others are virtually unchanged. It can be derived that **(a).** *VDs* of CA, F-F and SC varies in same charging snippets; **(b).** *VDs* of CA, F-F and SC evolve divergently in cycles as well. These are the reasons to choose *VD* as the key indicator to identify the failure modes in this work.

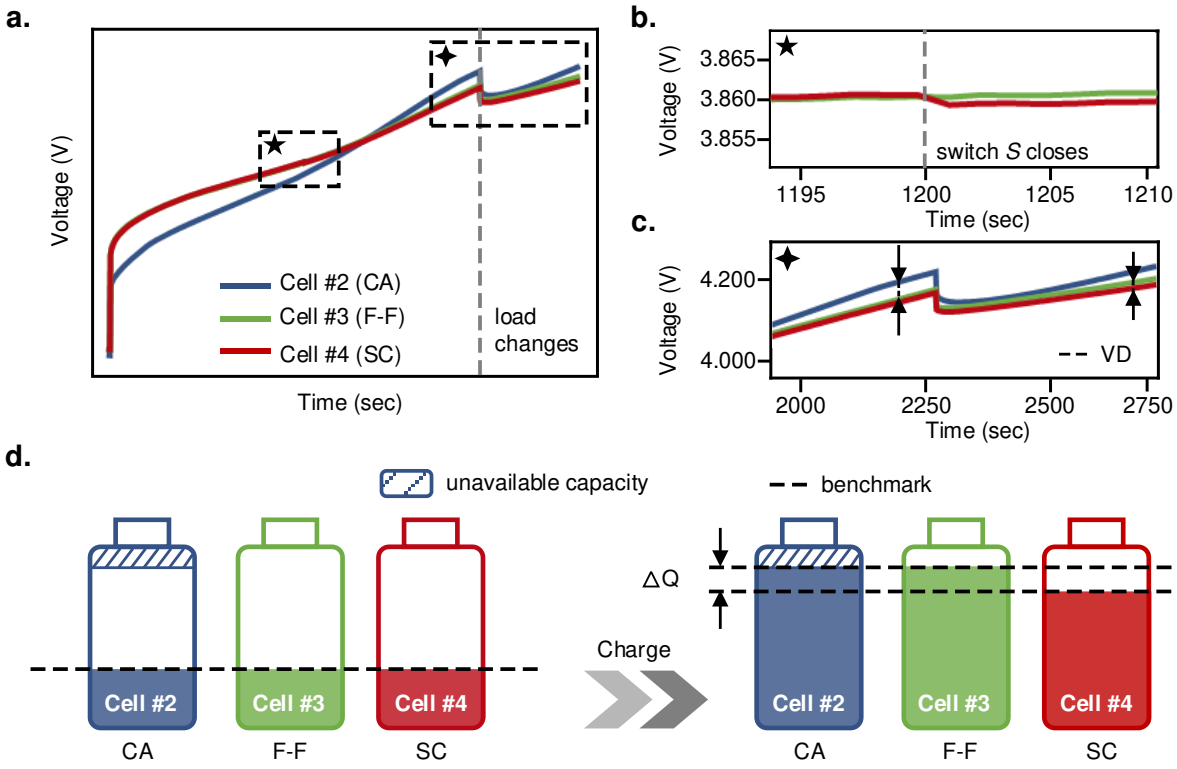

Fig.3 Distinct deterioration mechanisms of various fault types. a. Voltage responses during a multi-stage CC process; **b.** Local amplification of voltage drops at SC trigger; **c.** *VD* between CA cell, SC cell and FF cell; **d.** Charging capacity changes of different failure modes in series-connected battery system.

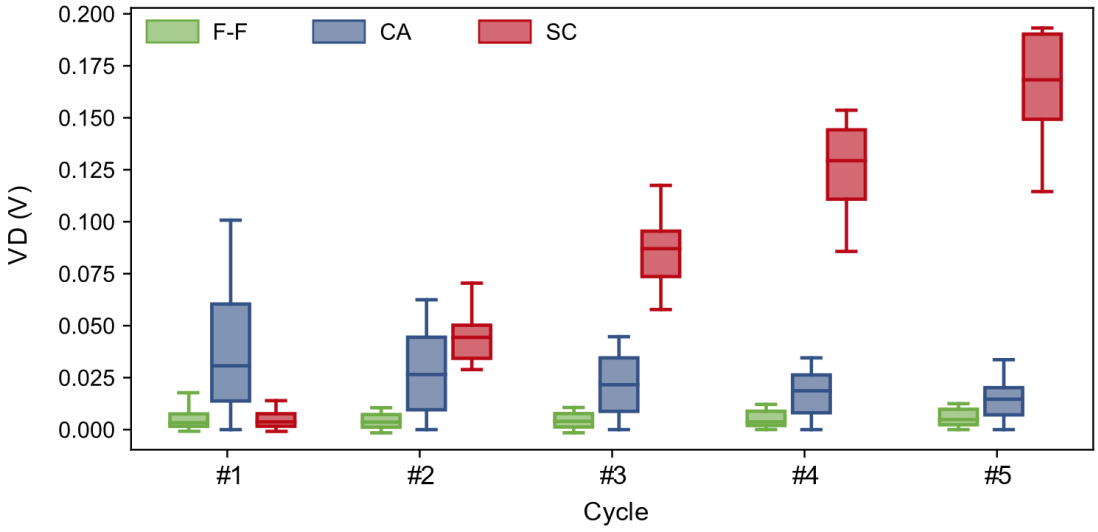

Fig.4 The evolutionary process of *VDs* of various fault types., in which subplots **#1**-**#5** represent the cycle 1-5 of the experiment, respectively.

3.3 Real-World EV Operational Data

In real-world EV application scenarios, due to the uncertainty caused by external environment, operating conditions, user behaviours, incidental events and so on, batteries may act differently than in a lab setting. To validate the feasibility of the proposed concept and models constructed by experimental datasets, we employ real-world EV operational data to further examine and optimize the overall diagnostic framework. These operational data are sourced from a typical EV model of the National Big Data Alliance of New Energy Vehicles (NDANEV) Open Lab, including normal vehicles and those involved in battery TR-induced accidents, is obtained for this part of the study. According to the standardized data transmission protocol, 73 data items, encompassing both vehicle and battery states, are collected to the platform in real-time with a sampling frequency of 0.1 Hz. The readers may refer to [35] for thorough particulars of data preprocessing and segment division. The detailed specifications of the studied vehicles are listed in **Table. 3**.

Table. 3 Specifications of the Studied EV Model

Items	Specifications
Curb Weight (kg)	1595
Pack Configuration	1p95s
Cell Cathode	NCM
Cell Capacity (Ah)	126
Cell Nominal Voltage (V)	3.65
The Battery Energy Capacity (kWh)	45

4 Methods

Theoretically, LIBs are complex and nonlinear electrochemical systems with a lack of measurable parameters in practical BMSs. This paper only utilizes measurable battery responses and the calculation of *VDs* of battery systems as the foundation to investigate the diagnostic strategies with the assistance of DL algorithms. With this simplification, the requirement for extra battery parameter identification and redundant parameter capturing is eliminated. Herein, we leverage the power of DL algorithms to transform cell responses into a time series problem, encompassing the electrochemical properties of the battery itself, the failure mechanisms and fault-induced variation in external characteristics across cells. Concretely, our diagnostic framework consists of three key sections: **(a)** Predicting future reference voltage responses at the cell level (4.1.1) and **(b)** Classifying multiple faults at the module level (4.1.2). **(c)** Determining the type of fault and quantifying the degree of the fault (4.2).

4.1 DL-Powered Diagnostic Algorithms

4.1.1 Reference Voltage Predictor

Terminal voltage is usually the most direct and rapid manifestation of the battery faults and abnormal states. Ideally, within a module, when the internal resistance and initial available remaining capacity of individual cells are consistent, the voltage profiles of the cells should coincide. Furthermore, when each cell is subjected to identical operational conditions and environments, their degradation trajectories should also be consistent, implying that the voltage profiles should exhibit the same trend of variation [36]. However, when some of the cells in the actual battery module are faulty, the voltage profiles of the faulty cells produce a different degree of outliers from the F-F cells.

Based on the above description, in this section we introduce the hypothesis of the reference voltage. The reference voltage in our diagnostic scheme demonstrates the standardized future voltage value of a battery cell. The predicted reference voltage values

enable further determination of whether an abnormality occurs by evaluating the actual measured voltage values. In Formulation **(1)**, we select an F-F cell as a reference to calculate the VDs and analyze the fault mechanism, however, the reference cell is hard to select in operating batteries. Thus, our goal is to predict the reference voltage in this section, providing a criterion of diagnostics for battery systems in advance. DL models are capable of automatically learning and extracting relevant features from time series data. This eliminates the need for manual feature engineering, making them more adaptable to a wide range of time series problems. In this paper, the recurrent neural network (RNN) kernel is used to establish the prediction model due to its strong capabilities in solving forecasting challenges in time series. Here we choose the gated recurrent unit (GRU) as the core layer of RNN. The

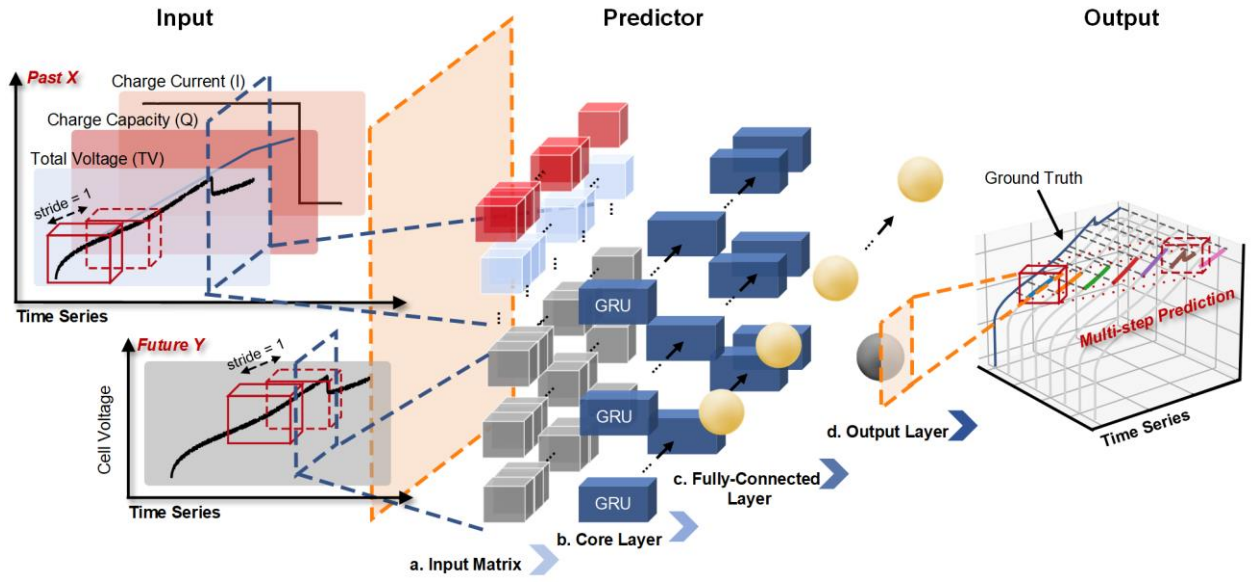

Fig.5. The modelling scheme illustration of the Predictor. The Predictor includes four main components: In **a**, the past *X* of the model input matrix involves a 500-second time window time series including total voltage (TV), charging current(I), charging capacity(Q) of the battery system and the corresponding future *Y* is the 500-second cell voltage response. Both time windows of *X* and *Y* samples every 1-second a stride. The core layer of the model consisting of a three-layer GRU for the forecasting task is provided in **b**. The information from previous layers is integrated in **c**, enabling the neural network to better understand complex relationships within the input data. **d**. The model output is the cell voltage with a 500-second sequence (same length as input *Y*) in this paper.

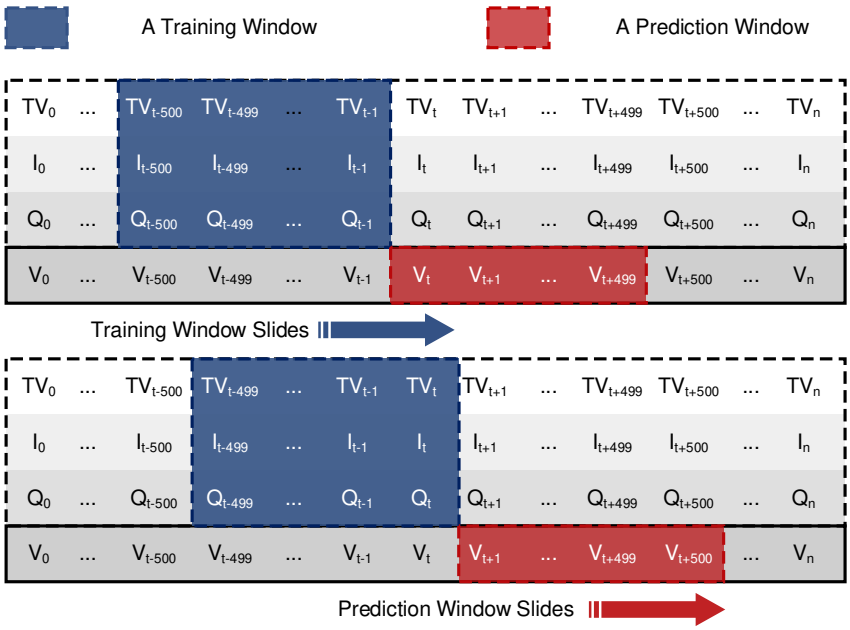

Fig.6 The processing of the input sequence battery data. Wherein, the blue (red) areas indicate a time window for the independent (dependent) variables for the training dataset, respectively. TV_t, I_t and Q_t represent the total voltage, charging current and charging capacity at the moment t; The movement for the time windows can be noted from the dashed box to the solid one. The method will be scanned from the beginning of the sequence to the end.

constructed GRU structure is displayed in **Fig.5** and the explanation of each applied layers are detailed in **Supplementary Note 2**. In addition, the time window is introduced for achieving multi-forward prediction and fully studying the information of the battery sequence, intercepting several 500-second battery sequence data with a 1-second stride to slide the time window as shown in **Fig.6**. In the Predictor, the past parameters collected from time series including charge current, charge capacity, total voltage and the corresponding terminal voltage of the battery system are used as the inputs of the model. The future reference voltage sequence is set as output.

4.1.2 Fault State Classifier

A hybrid algorithm, convolutional neural network (CNN) augmented with multi-head attention (CNN-mAtt), is utilized for the fault classification in selected charging snippets. As analyzed in Section 3.2, when multiple faults occur in battery systems at the same time, the VDs of different faulty cells exhibit variously and deteriorate at different rates. Thus, we select

VDs as the key fault indicator and then feed them into the Classifier as input. And the fault state labels (F-F, SC and CA) provided by experimental data is the output of the Classifier during the training phase, while a possibility of the fault state ($P_{F\text{-}F}$, P_{SC} and P_{CA}) will be finally predicted by the trained model. The construction of the Classifier model is sketched in **Fig.7** and the specific introduction of the algorithm and selected layers of the Classifier is demonstrated in **Supplementary Note 3**. Additionally, different from the Predictor, the Classifier requires one-hot encoding of their inputs for the data pre-processing (see **Supplementary Note 4**). These chosen algorithms have demonstrated their effectiveness in analyzing large datasets and are widely utilized. All proposed models are developed using the open-source neural network library *Keras* and the Python machine learning library *scikit-learn*.

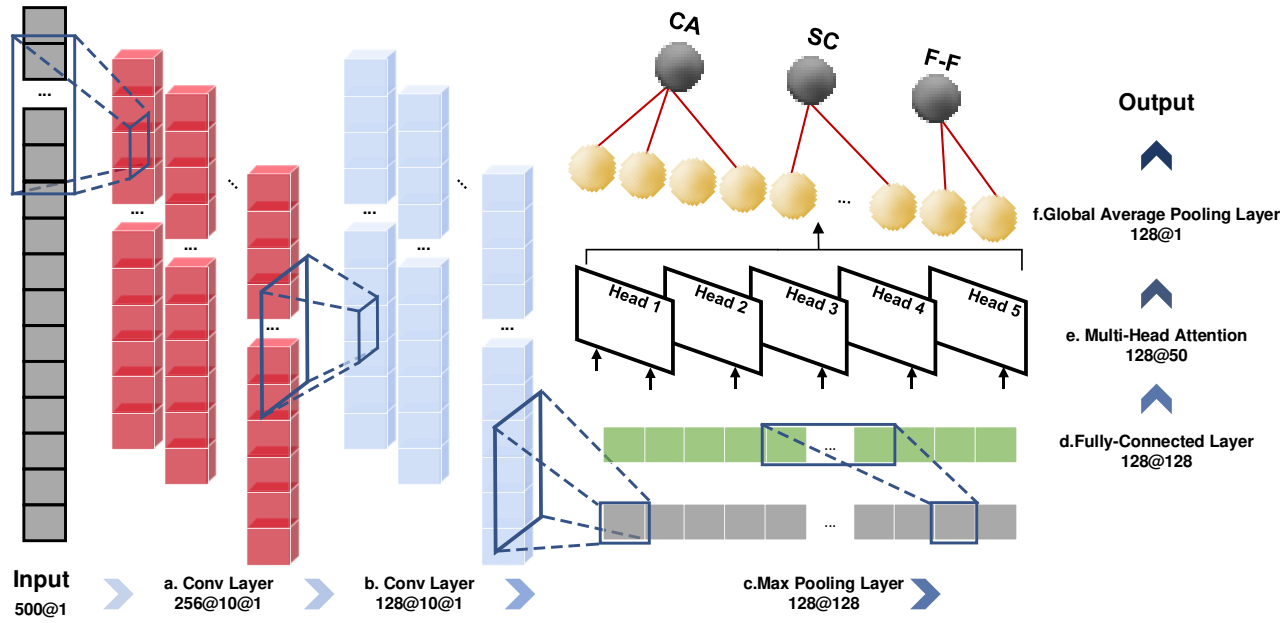

Fig.7 The modelling scheme illustration of the Classifier. The Classifier consists of four main components: Two kinds of convolutional layers are provided in **a** and **b**; The max pooling layer is shown in **c**, while the global average pooling layer is in **f**, respectively; The fully-connected layer is in **d**; The multi-head attention layer is in **e**.

4.2 Comprehensive Fault Identification and Quantification Strategy

As described previously, high-functioning DL models are constructed for reference voltage prediction and fault possibility classification. Still, misdiagnosis happens and cannot be completely avoided, given the occasional instability of the sensor-acquired data or changes in charging excitation. Herein, we try to provide a more reasonable and accurate solution to determine the fault type and fault degree. First, due to the contingent nature of each single diagnosis, we introduce evaluation windows and put into continuous results of the Predictor, where the diagnoses within a window will be again synthesized and evaluated. Then, the median for a certain number of probabilistic outcomes is calculated in each evaluation window to characterize the statistic assessment of the current window. The reason for choosing the median to characterize the statistical properties of each window is because of its robustness to outliers and extreme values. In addition, the results of the Classifier are presented in the form of probabilities and usually in machine learning multi-classification problems, the column with the largest probabilistic performance of classification results is considered to be more consistent with the current state. Therefore, in the second step of our strategy, we compare the three classes of probabilistic results and select the class with the largest probability to characterize the current fault state (F-F, CA or SC).

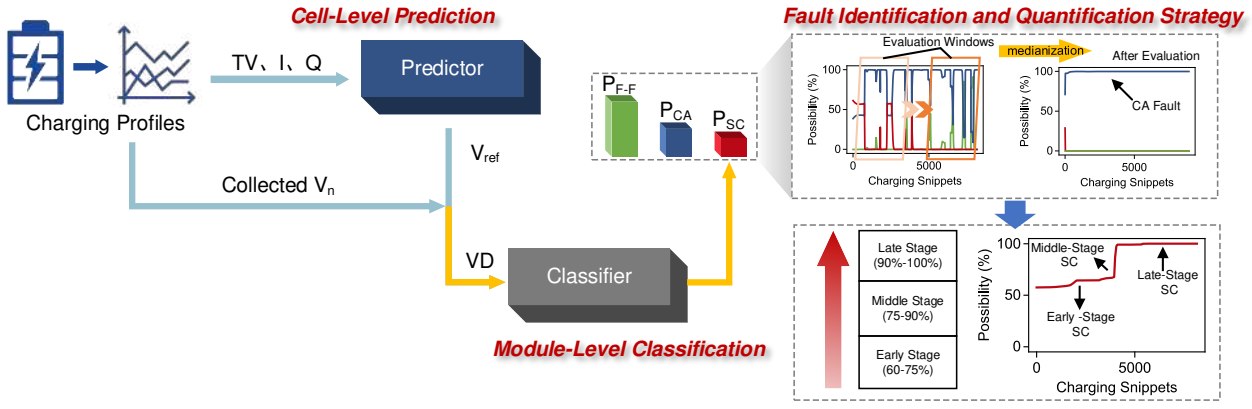

Fig.8 The overall workflow of the proposed fault identification and quantification strategy.

Once the type of fault has been identified, its corresponding probability will be used as an indicator to quantify its degree of failure. In this section, we artificially grade the degree of failure according to the median probability of the evaluation window, categorizing the severity of the failure into early-, middle- and late-stage. In our large number of observations, a probability of around 60% can be considered as a starting point for determining a failure, while 100% indicates a strong correlation. Thus, 60% to 75% (~75% to 90%, ~90% to 100%) is considered to be indicative of an early-stage (~middle-stage, ~late-stage) fault. The effectiveness of this grading strategy is also fully validated in the **Results and Discussion** section. In **Fig.8**, we summarily sketch the fault identification and quantification scheme of the paper as a whole, highlighting in particular the comprehensive fault identification and quantification strategy mentioned at the end.

5. Results and Discussions

We generate two separate experimental datasets, Group A and Group B, for training, validation and testing of all the models discussed. Indeed, we used the data from the first four cycles of Group A as the training dataset and the data from the last cycle of Group A as Test Dataset 1. To rigorously assess the performance of the Predictor and Classifier, instead of the traditional approach of splitting a portion of the original dataset for testing, we utilize an independent Testing Dataset 2 derived from the five-cycle data of Group B. This approach introduces more challenges for model construction but underscores the robustness of the proposed diagnostic scheme. In Section 5.1, 5.2 and 5.3, we mainly validate the effectiveness of the proposed method through the experimental dataset. Furthermore, we employ the real-world EV operational data for the applicability validation, demonstrating and enhancing the value of the practical application of the proposed framework.

The algorithm code was written in Python 3.10 and executed on a personal computer with a 13th Gen Intel(R) Core(TM) i5-13500H processor running at 2.60 GHz. The results of the diagnostic scheme are presented below:

5.1 Cell-Level Reference Voltage Prediction

Accurately predicting the reference voltage of each cell within a battery pack constitutes the primary objective of the diagnostic framework presented in this study. To this end, we harness the RNN for multi-step forward time series data prediction. Specifically, the GRU is chosen as the core layer in the RNN algorithm and the detailed structure of the prediction model is elucidated in the **Method** section. The proposed algorithm attains noteworthy performance metrics, with a mean square error (MSE) of 7.84×10^{-5} V, a mean average percentage error (MAPE) of 0.16% and an R square (R^2) value of 0.9, thereby manifesting a highly effective predictive capability. To underscore the advancements achieved by the finalized model, **Fig.9.** visually represents prediction results and compares the proposed method against other state-of-the-art algorithms. In pursuit of selecting the most suitable algorithm, a comprehensive analysis is conducted: **(a).** GRU structure comparison: Various GRU structures are compared to identify optimal parameters. We designate two additional GRU models as GRU1 and GRU2 (detailed in Supplementary Table S1). Among the three GRU structures depicted in **Figs. 9 a-e**, the structure exhibiting the most favorable overall performance is selected for the predictive model in this paper. This optimal model reveals errors smaller than 0.02 V in over 95% of the total samples. **(b).** Algorithm comparison: GRUs are juxtaposed with both common shallow learning algorithms and other DL algorithms. **Fig.9 g** illustrates that GRUs exhibit a lower MAE of approximately 0.0015 V, establishing their superiority compared to other state-of-the-art algorithms. Additionally, in **Table. 4**, we compare the prediction error of

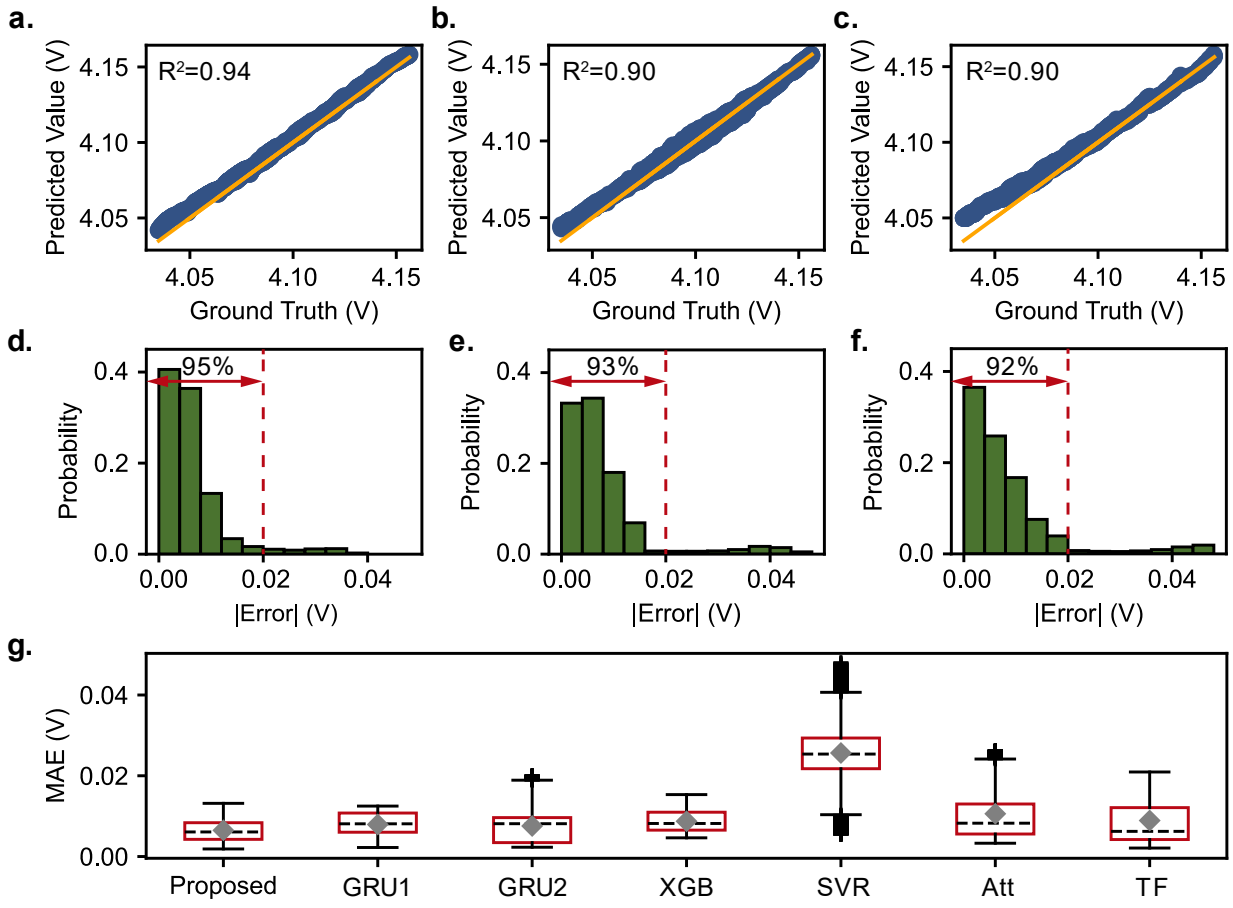

Fig. 9. Prediction results of a. Proposed algorithm**; b.** GRU1**; c.** GRU2**.** Prediction error probability density plots of d. Proposed algorithm; **e**. GRU1; **f**. GRU2. In **g**, we set out MAE boxplots of different state-of-the-art algorithms to present the superiority of our algorithm.

the proposed algorithm and other state-of-the-art algorithms including support vector regression (SVR), eXtreme Gradient Boost (XGB), attention (Att) and transformer (TF). The comparison of the computing time between the mentioned algorithms is illustrated in **Table. 5.** We adopt MSE, MAPE and R^2 as evaluation indicators (see **Supplementary Note 5**) to provide more detailed comparative results for both Test Dataset1 and Test Dataset2, offering a full validation of the proposed algorithm. Notably, Test Dataset2, being an entirely new dataset, yields slightly inferior model results, albeit within acceptable limits. Importantly, our proposed model consistently outperforms other evaluated algorithms across all metrics in both datasets.

Table. 4 Comparison of Prediction Errors of Different State-of-the-Art Algorithms.

Algorithms	Test Dataset1			Test Dataset2		
	MSE (V)	MAPE (%)	R^2	MSE (V)	MAPE (%)	R^2
Proposed	**7.84×10^{-5}**	**0.16**	**0.94**	**3.5×10^{-4}**	**0.32**	**0.95**
SVR	8.00×10^{-4}	0.63	0.42	1.52×10^{-3}	0.75	0.80
XGB	1.80×10^{-4}	0.21	0.87	6.43×10^{-3}	1.66	0.76
Att	2.12×10^{-4}	0.26	0.84	1.09×10^{-3}	0.47	0.86
TF	1.59×10^{-4}	0.22	0.89	1.17×10^{-3}	0.51	0.84
GRU1	1.36×10^{-4}	0.19	0.90	3.41×10^{-4}	0.36	0.92
GRU2	1.34×10^{-4}	0.19	0.90	3.03×10^{-4}	0.29	0.91

Table. 5 Comparison of Computing Time of Different State-of-the-Art Algorithms.

Algorithms	Computing Time (s)	
	Training	Testing
Proposed	**1,765.35**	**3.82**
SVR	1120.60	9.98
XGB	10,329.71	10.19
Att	5956.05	2.34
TF	1,766.58	1.96

5.2 Module-Level Fault Classification

After obtaining reference voltage responses for individual cells through a constructed Predictor, a CNN-mAtt is employed for the classification of cell states (F-F, SC and CA) to determine the safety of the battery. *VDs* are chosen as the model input due to their superior sensitivity to faults and relevance to underlying mechanisms, effectively representing distinctions among cells within a battery pack for Predictor training. Our Classifier demonstrates commendable performance after the appropriate process of training. Notably, the Classifier achieves a prediction accuracy of 91.96% and 89.20% on Test Dataset1 and Test Dataset2, respectively. The predictive performance of each category presented by

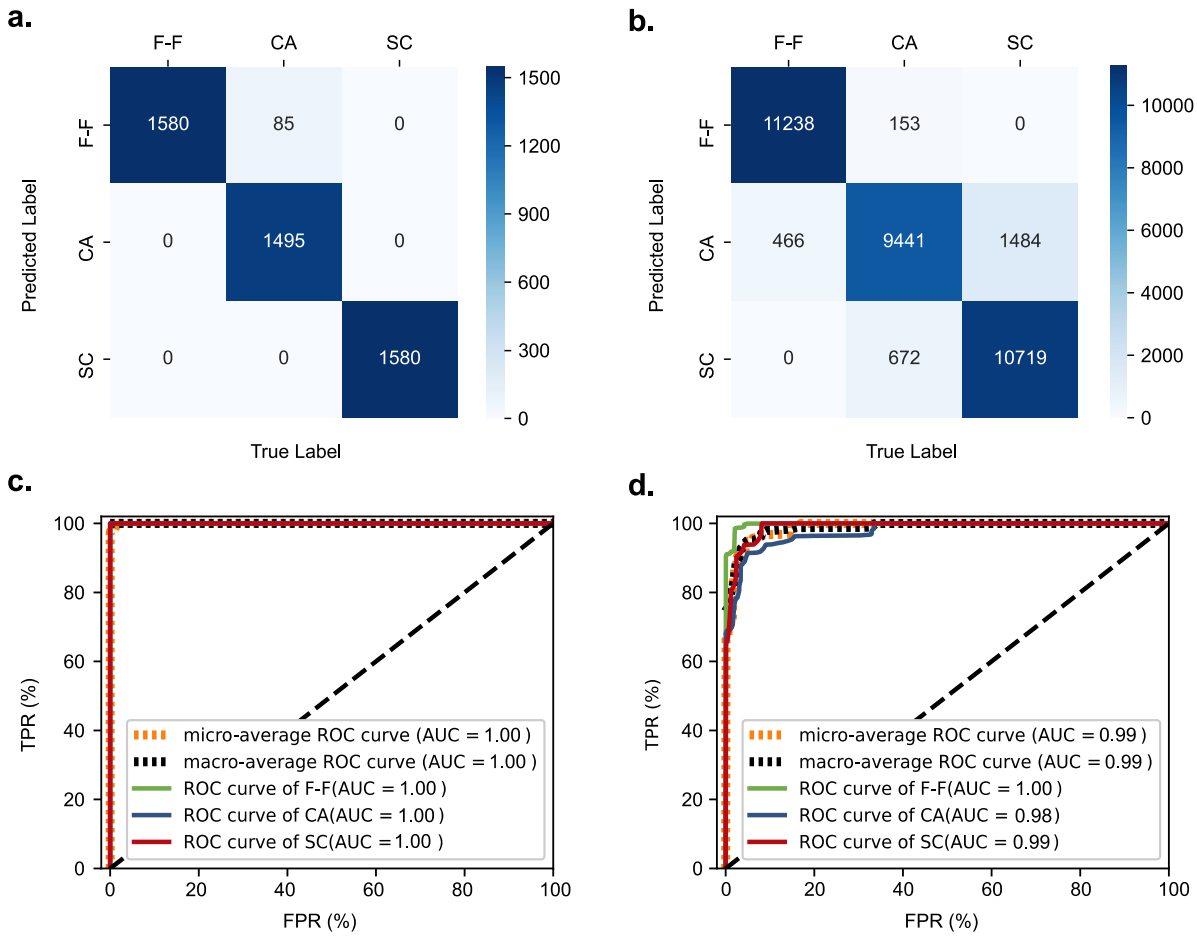

Fig.10 Classification results of the proposed Classifier. Confusion matrix **a.** shows the predicted effect on Test Dataset1 and each category involves 1,580 samples while **b.** Test Dataset2 contains 11,391 samples. ROC curves of the Classifier are demonstrated on **c.** Test Dataset1 and **d.** Test Dataset2, respectively.

confusion matrix is detailed in **Figs. 10 a-b**. **Figs.10 c-d** illustrate the receiving operating characteristics (ROC) curve of the Classifier, with the area under the curve (AUC) representing the discriminative performance across various classes (see **Supplementary Note 6**). In addition, we calculate micro-average ROC and macro-average ROC for a clearer insight into the model performance. The AUC results for each category demonstrate the strong classification ability of the proposed model, with proceeding 96% for each category both on Test Dataset1 and Test Dataset2. This classification capability holds promise for practical applications in BMSs of EVs, facilitating accurate identification of LIB fault types.

To further validate the excellence and robustness of our classification model, as in the previous section, we use more algorithms (including CNN, ATT, TF, XGB and support vector machine (SVM)) and test them on Test Dataset1 and Test Dataset2 for comparison (displayed in **Table. 6)**. We prefer to choose the model that all classification results are relatively good evenly. Significantly, our proposed model is in the lead in comparison with other models. The proposed model performs a 91.7% accuracy in Test Dataset2, while there shows an accuracy of 98.7%, 82.0% and 94.1% in three categories, respectively. Different from the prediction model, it should be noted that this section is a multi-classification task and on top of the accuracy of each class separately that we use as an evaluation metric. It is also necessary to comprehensively consider the balance of the accuracy of results of three classes. If only accuracy results of individual classes are good among all the results, it will instead indicate that the model has a high false alarm rate. Moreover, although the comprehensive accuracy of TF in Test dataset2 reaching 80.2%, its classification accuracy in the third category can only be 66%. This result means that there is still a high probability of predicting SC faults as F-F or CA, which brings a negative impact on improving the misdiagnosis rate of the model. Besides, the comparison of computing time of different state-of-the-art algorithms is illustrated in **Table. 7**. It can be observed that the proposed model

requires more training time compared to most of the comparison models due to the increased number of network layers. However, the testing time marginally exceeds that of the other models. It is worth noting that the proposed battery diagnostic framework is designed to be trained in a cloud environment and implemented in onboard BMSs. In this context, the training time is not a significant concern, and the proposed model remains feasible for practical application.

Table 6. Comparison of Classification Accuracy of Different State-of-the-Art Algorithms.

Algorithms	Test Dataset1				Test Dataset2			
	Acc (%)	Acc1 (%)	Acc2 (%)	Acc3 (%)	Acc (%)	Acc1 (%)	Acc2 (%)	Acc3 (%)
Proposed	**98.2**	**100.0**	**94.6**	**100.0**	**91.7**	**98.7**	**82.0**	**94.1**
CNN	88.3	85.3	90.2	89.4	80.5	89.7	76.6	75.1
Att	91.3	93.4	92.5	87.9	85.6	90.2	86.7	79.8
TF	90.8	89.6	92.5	90.3	80.2	85.6	88.9	66.0
XGB	86.9	99.8	61.1	100.0	75.2	82.9	59.2	83.4
SVM	91.1	100.0	73.2	100.0	83.4	94.9	81.4	73.8

NOTE: Acc, Acc1, Acc2, Acc3 demonstrate the accuracy in all of the samples, the first (F-F), the second (CA) and the third (SC) category of samples, respectively.

Table. 7 Comparison of Computing Time of Different State-of-the-Art Algorithms.

Algorithms	Computing Time (s)	
	Training	Testing
Proposed	**5,671.31**	**8.91**
CNN	4,982.62	7.69
SVM	7320.89	7.22
XGB	3480.14	5.83
Att	1,616.10	2.44
TF	6,921.42	8.66

5.3 Comprehensive Safety Evaluation Strategy

To ensure the practical application capability of the overall diagnostic framework, we make contributions to minimize potential misdiagnoses and present a comprehensive fault identification and quantification strategy. Based on the superior performance of the Predictor and Classifier, we further take both the fault type determination and the degree of its corresponding safety hazards into account. Herein, we first adopt evaluation windows by

putting together a period of initial diagnosis results. Then take the median of the data in the evaluation windows to characterize and evaluate the results of a range of assessments statically. In this section, we focus on the validity of the proposed evaluation strategy and carry out the validation on the Test Dataset2. We apply the strategy on an F-F cell, a CA cell and an SC cell and check the evaluation performance on evaluation windows of different length sizes to determine a better choice of window size.

In **Fig. 11**, the three subplots in each row represent F-F, CA and SC cells and each column represents a window size of 1, 1,000 and 2,000, respectively. As observed, with the increase of the diagnostic window, the frequency of misdiagnosis decreases significantly, especially at a window size of 2,000 and misdiagnosis is almost non-existent in our dataset. Thus, it is quicker to determine the type of fault and the greater the length of the evaluation window, the lower the frequency and the less time consumption required for the evaluation. In addition, the quantification efficiency of the proposed evaluation strategy can be strongly proved. **Fig.11 c** presents a very high $P_{F\text{-}F}$, indicating the F-F; In **Fig.11 f.** the CA cell is first diagnosed as middle-stage CA and then becomes late-stage CA. In our strategy, we consider the probability at this point to be the severity of the fault. It can also be seen in **Fig.11 i** that our evaluation results of P_{SC} consider the cell to be an early-stage at the beginning of the cycle; while the probability grows as the number of cycles goes deeper, gradually becoming middle-stage and late-stage. This not only realizes the accurate identification of faults but also matches the actual fault deterioration mechanism of the cell in terms of the quantification of degree of faults.

It should be emphasized that the evaluation window does bring more requirements for input data but also a better diagnostic performance. The size of the evaluation window can be adjusted by the practical application scenario and the tolerability because even under the window size of 1, the misdiagnosis rate performed no more than 10%.

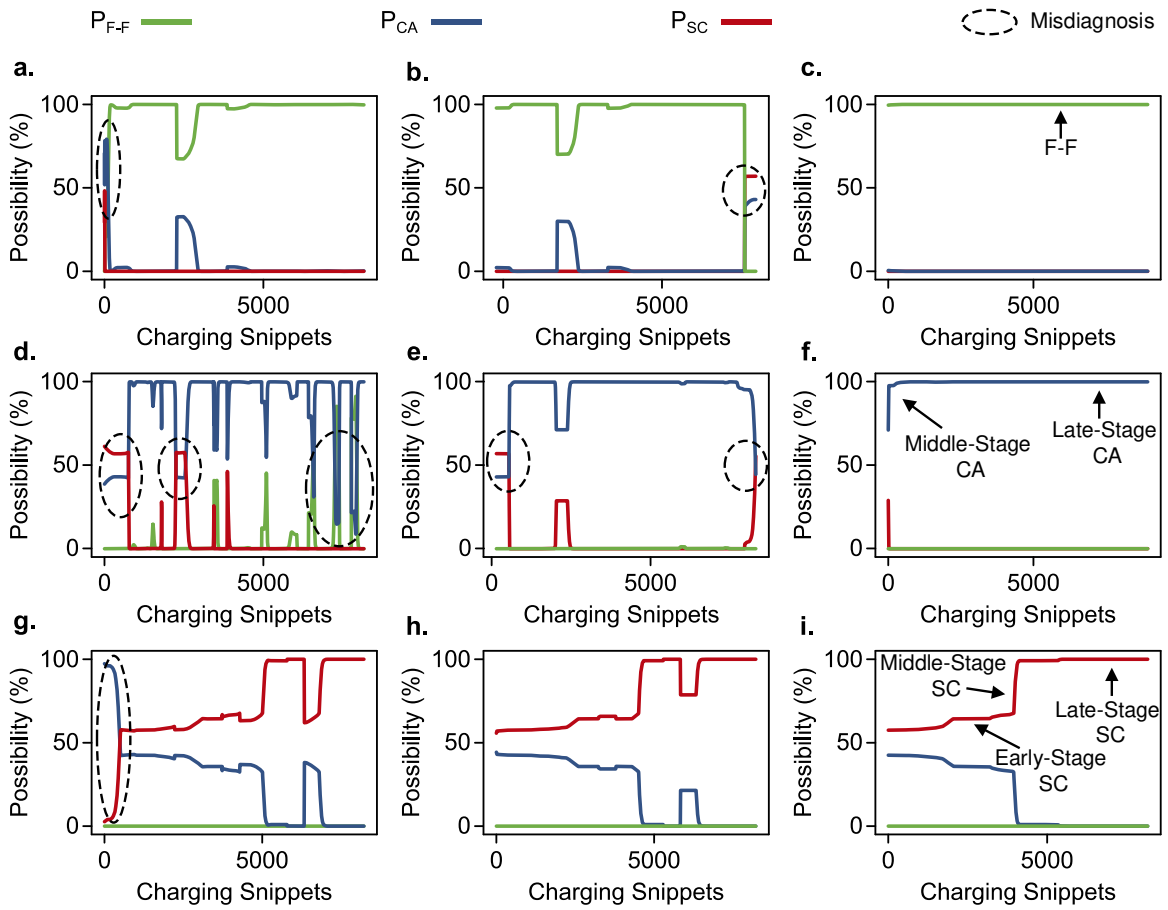

Fig.11 Safety Evaluation results of different cells. a.-c. display possibility results of an F-F cell. **d.-f**. are results of a CA cell; **g.-i.** are the performance of the SC cell based on the proposed strategy. Each column of subfigures represents an evaluation window of size 1, 1,000 and 2,000, respectively.

5.4 Diagnostic Framework Feasibility in Real-World Vehicles

To further prove the practical applicability of the proposed diagnostic framework based on experimental datasets, we introduce real-world vehicle data collected from NDANEV to comprehensively validate the entire technical framework of diagnostics, including reference voltage prediction and battery fault state classification. The diagnostic effectiveness of the proposed diagnostic framework in real operating vehicles is presented in this section. The detailed information of the applied vehicle model is demonstrated in Section 3.3. **Fig.12** provides a general picture of real-world cloud EV data. **Figs.12 a.-c.** show the basic collection items from the cloud data platform including voltage, current and temperature profiles of

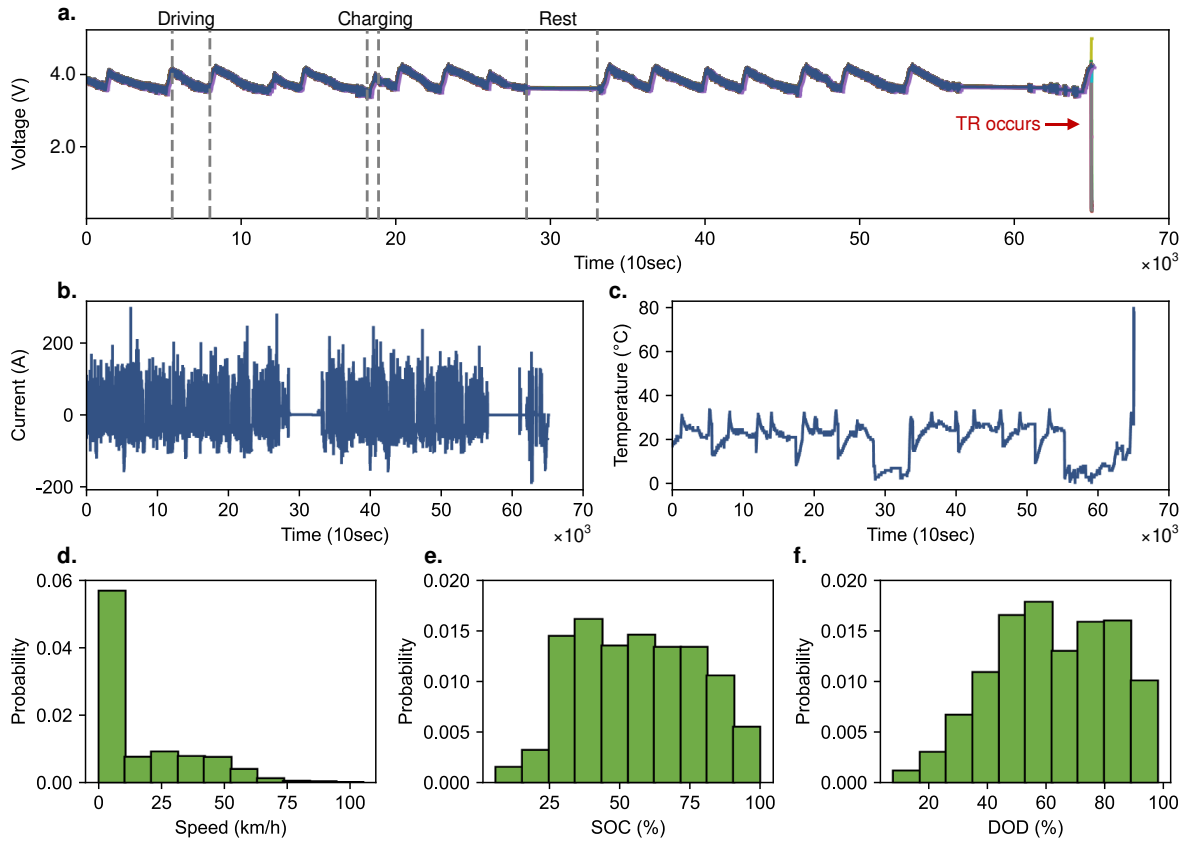

Fig.12 Illustration of the collected real-world cloud dataset. a. is an example clip of an accident vehicle before the TR occurred, including driving, charging and rest process. **b.** and **c.** are the current and temperature profiles. **d.- f.** are speed, SOC and DOD distribution of the typical vehicles.

operational data of an accident vehicle as an example. In addition, speed, SOC and depth of discharge (DOD) have been statistically analyzed in **Figs.12 d. - f.** for a clear understanding of the selected vehicle model.

Notably, in our previous diagnostic modeling based on experimental data, we utilized all parameters directly obtainable from the vehicle, highlighting a key strength of our study. Consequently, for real-world vehicle validation, no additional sensors are required, nor are any extra calculations or transformations on the parameters necessary. Meanwhile, the core layers of all models and the overall framework remain consistent with those described previously. However, due to the different data collection frequencies in the laboratory (1Hz) and the real-world vehicle (0.1Hz), we adjust the size of the sliding window selection for the

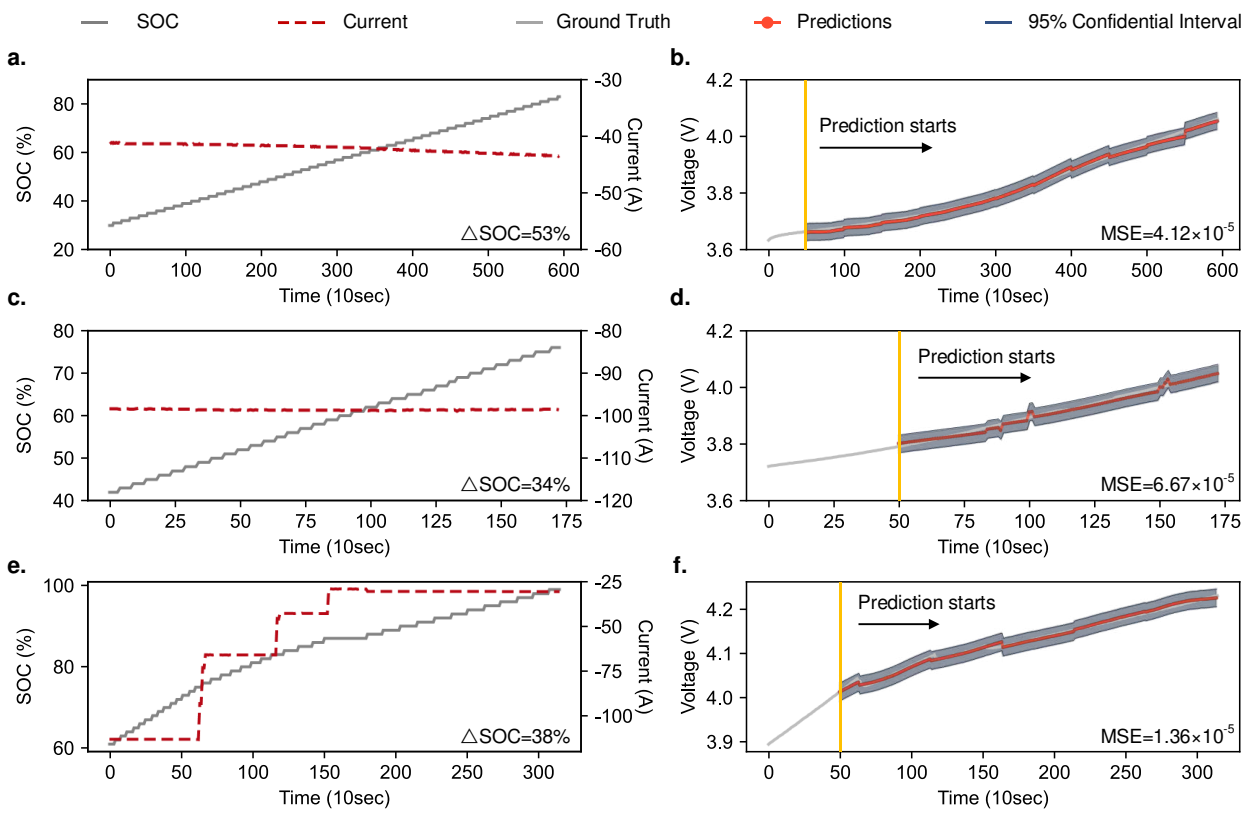

Fig.13 Reference voltage prediction results under real-world multi-charging scenarios. a., c., and e. are the current and SOC profiles during CCSC, CCFC and MSCC charging modes. **b., d. and f.** are the corresponding prediction results under the mentioned charging scenarios.

real-world vehicle. The original sliding window containing 500 frames of data is reduced to a 50-frame sliding data window to ensure the data covered by the window has the same time scale.

In the real-world validation of reference voltage prediction, current, SOC, and TV of the battery module obtained from charging segments are selected as inputs for the Predictor, with the future reference voltage of individual cells as the model output. To present comprehensive demonstrations of the model performance, we have conducted validation around the three most common EV charging scenarios including constant current slow charging (CCSC), constant current fast charging (CCFC), multi-stage constant current charging (MSCC) and differentiated SOC starting and stopping ranges. As shown in **Fig.13**, the Predictor performs well in all three charging scenarios. Concretely, in **Figs.13 a.- b.**, the selected CCSC charging snippets charge from ~30% to ~80% SOC, which can be regarded

as a most fluent operational distribution of SOC [1]. Besides, as illustrated in **Figs.13 c. – f.** model performance under CCFC and MSCC modes are evaluated as well. And the MSE under CCSC, CCFC and MSCC reach 4.12×10^{-5}, 6.67×10^{-5} and 1.36×10^{-5}, respectively, proving the usability of our model in most common cases under actual applications.

Next, a faulty vehicle is employed to validate the Classifier and the proposed safety evaluation strategy. The selected vehicle was dismantled after the accident and it was confirmed that the cause of the accident was due to battery failures. In this section, the inputs and outputs of the Classifier are all consistent with the previous description. The proposed model performs a 97.4% accuracy in the cloud EV dataset, while there shows an accuracy of 98.5%, 96.3% and 97.3% in three categories (F-F, CA and SC), and the corresponding ROC curves and confusion matrix are shown in **Figs.14 a–b.** Particularly, practical effects of the overall diagnostic frameworks are considered and discussed. Among the 95 cells that make up the battery system, two of which (Cell #83 and Cell #24) are suspected to have CA and SC faults. **Fig.14 c.** is the voltage profiles of the charging snippet three-cycle earlier than the accident happens. It can be seen that fault characteristics are not directly visible to the naked eye due to it is in the beginning of the fault and is very insidious. **Fig.14 d.**, is the last charging snippet before the TR occurs and the fault evolution is obvious. Herein, we use three-cycle ago of TR data for the validation and the results are shown in **Figs.14 e.-g.** It is worth to note that we employ evaluation windows of 100-frame and 60% of probability is considered as the alarm threshold. Specifically, **Fig.14 e.** is the evaluation result of F-F cells, and the $P_{F\text{-}F}$ stay near 100%, demonstrating that our diagnostic framework can give accurate judgments about the state of normal cells. **Fig.14 f.** shows the evaluation result of Cell #83, the P_{CA} starts to exhibit an increasing trend after the 200th evaluation window and finally alarms at the 230th window. Similarly, P_{SC} of Cell # 24 in **Fig.14 g.** alarms at the 165th evaluation window. Detailed alarm moments are also marked in the **Fig.14 c.** for a more intuitive representation. Overall, the proposed diagnostic framework can achieve accurate

prediction of future reference voltages while also providing precise assessment of the fault state of the battery system.

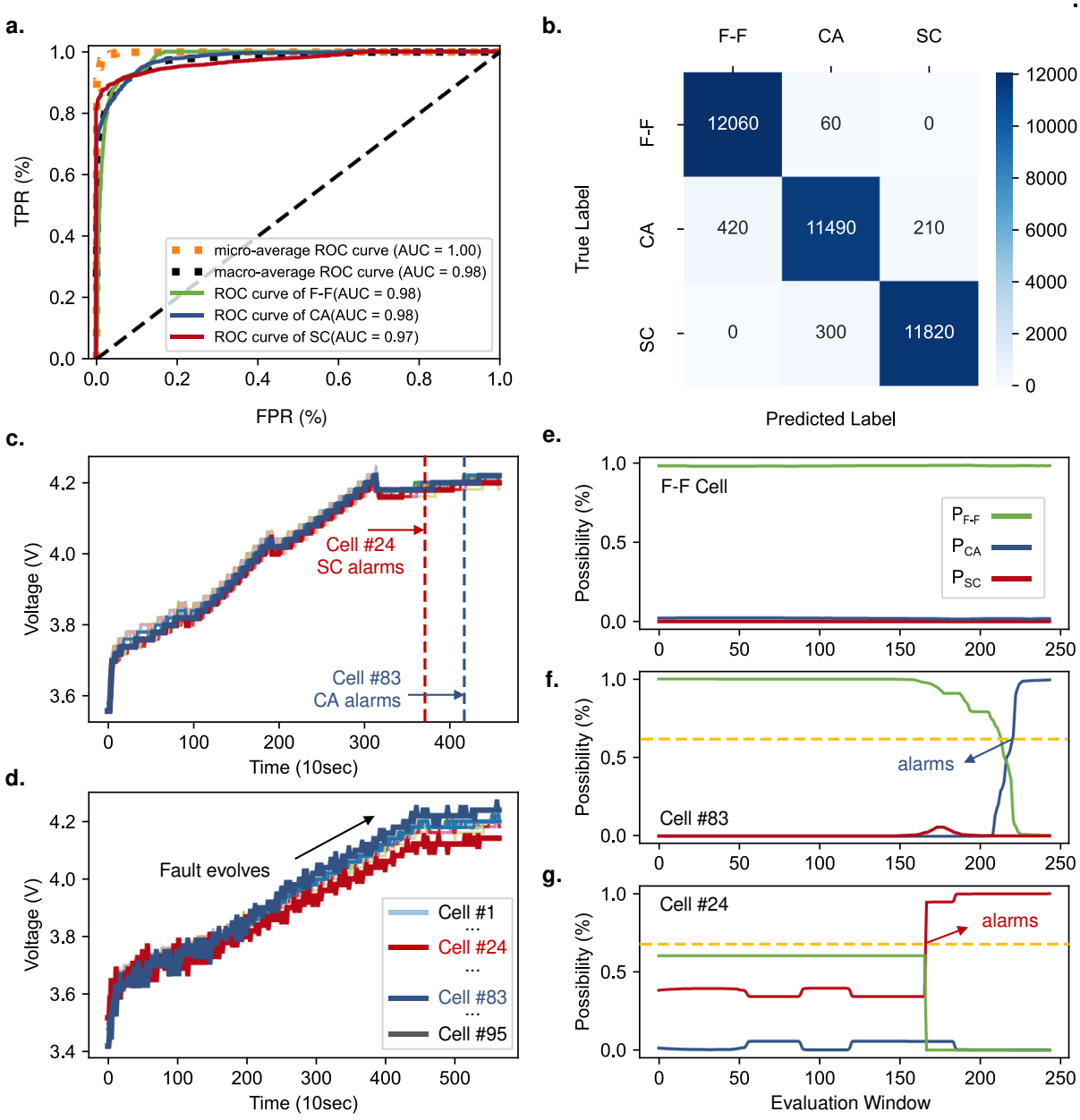

Fig.14 Diagnostic results of potentially faulty vehicles. a. ROC curves; **b.** Confusion matrix. **c.** is the voltage profile of the charging snippet three-cycle earlier than the accident. **d.** The last charging snippet before the accident. **e.- g.** are the diagnostic results based on c. mentioned charging snipped for the classes of F-F, CA and SC, respectively.

5.5 Comparison of other methods

The diagnostic framework proposed in this paper mainly addresses the multiple fault detection and quantification utilizing DL-based algorithms. Still, there exist methods do not rely on artificial intelligence and battery modelling. Here, the diagnostic framework proposed in this paper is compared with other fault detection methods in terms of computational complexity, fault type, accuracy and capability of implementation.

5.5.1 Correlation-Based Method

Xia *et al.* [37] innovatively adopted Pearson correlation coefficient (PCC) to detect SC faults in 2016. This type of method compares the correlation coefficients of the voltages of adjacent cells in a series-connected battery module, and determines the location of the SC cell by the abnormal correlation coefficients. Later, in recent research work of multiple faults diagnosis, correlation-based method is widely applied to detect the existing faults in the battery systems. Specifically, researchers [38–40] design redundant interleaved voltage measurement topology to detect the fault signal by calculating correlation relationships between cells, which is effective but hard to apply to real-world BMS due to the challenges for data collection and complicated to compute. The proposed diagnostic scheme directly adopts battery responses provided by sensors for the overall technical process, which improves the capability and timeliness for practical application with no need for additional measuring hardware.

5.5.2 Unsupervised Learning-based Method

Unsupervised learning is commonly based on limited data and considers sample points that differ from normal data as anomalies, making it difficult to distinguish between faults when data is generated by more fault types without pre-training. Besides, faults are usually extremely insidious in the early stages of their occurrence and behave similarly to F-F cells, making them difficult to diagnose by looking for outliers. Compared to the unsupervised

learning-based diagnosis scheme [41], the proposed supervised learning-based method provides a labeled insight to practical application due to the interpretable experiments.

5.5.3 Threshold-based Method

Some researchers consider anomalies in acquired signals as indicators of battery faults. To diagnose outliers in a battery module, methods such as calculating entropy, assessing curve similarity, and setting thresholds are employed [42,43]. Zhao *et al*.[24] have deployed rate of degradation and divided threshold to whether the capacity of battery cells are degraded abnormally. However, faults like sensor malfunctions and connection issues can also manifest as anomalies when triggered. Consequently, relying solely on abnormal behavior parameters for fault diagnosis is neither comprehensive nor reliable, often leading to misdiagnosis. Additionally, threshold-based methods struggle to accurately identify the specific type of fault. In contrast, the classification algorithm proposed in this paper enables precise identification of fault characteristics through the prior training of labeled data.

5.5.4 Model-based Method

Model-based fault diagnosis method compares estimated values from battery models with actual measurements. The difference, or residual, serves as a fault signal, which helps analyze fault characteristics and supports fault diagnosis. Reference [44,45] use a mean difference model and extended Kalman filtering to estimate the difference between the individual cell SOC and the average SOC in a battery pack. This difference is then used with a recursive least squares filter to form a method for diagnosing micro SC in batteries, allowing for accurate detection of SC resistance. However, the authors only discuss identifying fault sources using experimental data under adjacent charging and discharging conditions, which may not always be available in real-world EVs. Therefore, model-based methods have stricter requirements for the form, quality, and accuracy of the data. Using directly measurable parameters is more straightforward and less complex.

The summary of the comparisons of the mentioned fault detection methods are illustrated in **Table. 8.** for more details.

Table 8. Comparison of Fault Detection Methods.

Detection Method	Battery Model	Accuracy	Computational Complexity	Fault Type	Implementation
Correlation-based[38–40]	No	High	High	No	No
Unsupervised Learning-based[41]	Yes	Low	High	No	Yes
Threshold-based[42,43]	No	Low	High	No	Yes
Model-based[44,45]	Yes	High	High	No	No
Proposed	**Yes**	**High**	**Low**	**Yes**	**Yes**

6. Conclusions

This paper proposes a DL-powered multi-fault diagnostic scheme for series-connected battery systems. First, we carry out series-connected cycling battery experiments while injecting the two most common electrical faults including CA fault and SC fault concurrently within the circuit. By observing the external characteristics of cells of different faults and comparing them with F-F cells, the fault mechanism and its evolutional processes of various faults are further analyzed, providing a solid foundation for the following fault feature selection. Subsequently, our paper employs DL-powered algorithms, including an RNN-based Predictor and a CNN-based Classifier. These algorithms are designed to predict the reference voltage at the cell level while categorizing cells at the module level. The Predictor is trained with present TV, I and Q as inputs and future V as output, achieving an impressive MSE of 7.84×10^{-5} V. Meanwhile, the Classifier, trained with *VDs* as input and probability of fault state as output, attains an accuracy of 98.2%. The efficacy and superiority of our proposed models are extensively validated across diverse datasets, outperforming other state-of-the-art algorithms. Furthermore, we develop a comprehensive fault identification and quantification

strategy. This strategy involves applying evaluation windows and statistical tools to determine the fault type. Subsequently, the probability values corresponding to the identified fault types are utilized to quantify the degree of failure. Specifically, probabilities falling within 60%-75% are categorized as early-stage faults, 75%-90% as middle-stage faults and those ranging from 90% to 100% as late-stage faults. In order to further validate the implementability of the proposed diagnostic method, we introduce the cloud real-world EV data from NDANEV. The validation results show that the proposed models enable effective multiple fault detection in various practical operational scenarios. Consequently, the technical framework presented in this paper aims to enable the effective diagnosis of multi-faults in battery systems, thereby enhancing the active safety capabilities and prolonging the lifespan of EVs.

While our present work contributes significantly, it is essential to acknowledge certain limitations. On the one hand, we recognize that temperature is an important factor that affects both the internal and external performance of LIBs, representing the thermal characteristics as well. All of the experiments in this paper are conducted under controlled indoor temperature with the ignorance of diverse ambient conditions. Exploring the coupling effect of temperature gradients and the injection of multiple faults into LIB systems presents an avenue for valuable research. On the other hand, the proposed fault diagnosis scheme is constructed for individual types of LIBs (cylindrical ternary cells with $Li(NiCoAl)O_2$ cathode and graphite anode) and thus the generalizability to other types of LIBs is not explicit.

Acknowledgement

This work is supported by the National Key R&D Program of China under Grant 2022YFE0207800 and the National Natural Science Foundation of China under Grant U21A20170.

References

[1] Cui D, Wang Z, Liu P, Wang S, Zhang Z, Dorrell DG, et al. Battery electric vehicle usage pattern analysis driven by massive real-world data. Energy 2022;250:123837.

[2] Zhao J, Feng X, Tran M-K, Fowler M, Ouyang M, Burke AF. Battery safety: Fault diagnosis from laboratory to real world. Journal of Power Sources 2024;598:234111. https://doi.org/10.1016/j.jpowsour.2024.234111.

[3] Feng X, Ren D, He X, Ouyang M. Mitigating Thermal Runaway of Lithium-Ion Batteries. Joule 2020;4:743–70. https://doi.org/10.1016/j.joule.2020.02.010.

[4] Zhou L, Zhang Z, Liu P, Zhao Y, Cui D, Wang Z. Data-driven battery state-of-health estimation and prediction using IC based features and coupled model. Journal of Energy Storage 2023;72:108413. https://doi.org/10.1016/j.est.2023.108413.

[5] Heenan TMM, Mombrini I, Llewellyn A, Checchia S, Tan C, Johnson MJ, et al. Mapping internal temperatures during high-rate battery applications. Nature 2023;617:507–12. https://doi.org/10.1038/s41586-023-05913-z.

[6] Li W, Zhu J, Xia Y, Gorji MB, Wierzbicki T. Data-Driven Safety Envelope of Lithium-Ion Batteries for Electric Vehicles. Joule 2019;3:2703–15. https://doi.org/10.1016/j.joule.2019.07.026.

[7] Cui D, Wang Z, Liu P, Zhang Z, Wang S, Zhao Y, et al. Coordinated Charging Scheme for Electric Vehicle Fast-Charging Station with Demand-based Priority. IEEE Transactions on Transportation Electrification 2023:1–1. https://doi.org/10.1109/TTE.2023.3334809.

[8] Finegan DP, Billman J, Darst J, Hughes P, Trillo J, Sharp M, et al. The battery failure databank: Insights from an open-access database of thermal runaway behaviors of Li-ion cells and a resource for benchmarking risks. Journal of Power Sources 2024;597:234106. https://doi.org/10.1016/j.jpowsour.2024.234106.

[9] Deng J, Bae C, Denlinger A, Miller T. Electric Vehicles Batteries: Requirements and Challenges. Joule 2020;4:511–5. https://doi.org/10.1016/j.joule.2020.01.013.

[10] Schöberl J, Ank M, Schreiber M, Wassiliadis N, Lienkamp M. Thermal runaway propagation in automotive lithium-ion batteries with NMC-811 and LFP cathodes: Safety requirements and impact on system integration. eTransportation 2024;19:100305. https://doi.org/10.1016/j.etran.2023.100305.

[11] Finegan DP, Zhu J, Feng X, Keyser M, Ulmefors M, Li W, et al. The Application of Data-Driven Methods and Physics-Based Learning for Improving Battery Safety. Joule 2021;5:316–29. https://doi.org/10.1016/j.joule.2020.11.018.

[12] Li X, Gao X, Zhang Z, Chen Q, Wang Z. Fault Diagnosis and Detection for Battery System in Real-World Electric Vehicles Based on Long-Term Feature Outlier Analysis. IEEE Trans Transp Electrific 2024;10:1668–79. https://doi.org/10.1109/TTE.2023.3288394.

[13] Tian J, Chen C, Shen W, Sun F, Xiong R. Deep Learning Framework for Lithium-ion Battery State of Charge Estimation: Recent Advances and Future Perspectives. Energy Storage Materials 2023;61:102883. https://doi.org/10.1016/j.ensm.2023.102883.

[14] Liu X, Ren D, Hsu H, Feng X, Xu G-L, Zhuang M, et al. Thermal Runaway of Lithium-Ion Batteries without Internal Short Circuit. Joule 2018;2:2047–64. https://doi.org/10.1016/j.joule.2018.06.015.

[15] Mei W, Zhang L, Sun J, Wang Q. Experimental and numerical methods to investigate the overcharge caused lithium plating for lithium ion battery. Energy Storage Materials 2020;32:91–104. https://doi.org/10.1016/j.ensm.2020.06.021.

[16] Guo Z, Yang S, Zhao W, Wang S, Liu J, Ma Z, et al. Overdischarge-induced evolution of Cu dendrites and degradation of mechanical properties in lithium-ion batteries.

Journal of Energy Chemistry 2023;78:497–506. https://doi.org/10.1016/j.jechem.2022.12.013.
[17] Qiao D, Wei X, Jiang B, Fan W, Lai X, Zheng Y, et al. Quantitative Diagnosis of Internal Short Circuit for Lithium-Ion Batteries Using Relaxation Voltage. IEEE Trans Ind Electron 2024:1–10. https://doi.org/10.1109/TIE.2023.3342289.
[18] Yu Q, Dai L, Xiong R, Chen Z, Zhang X, Shen W. Current sensor fault diagnosis method based on an improved equivalent circuit battery model. Applied Energy 2022;310:118588.
[19] Karger A, Schmitt J, Kirst C, Singer JP, Wildfeuer L, Jossen A. Mechanistic cycle aging model for the open-circuit voltage curve of lithium-ion batteries. Journal of Power Sources 2024;593:233947. https://doi.org/10.1016/j.jpowsour.2023.233947.
[20] Kong X, Zheng Y, Ouyang M, Lu L, Li J, Zhang Z. Fault diagnosis and quantitative analysis of micro-short circuits for lithium-ion batteries in battery packs. Journal of Power Sources 2018;395:358–68. https://doi.org/10.1016/j.jpowsour.2018.05.097.
[21] Qiao D, Wei X, Fan W, Jiang B, Lai X, Zheng Y, et al. Toward safe carbon–neutral transportation: Battery internal short circuit diagnosis based on cloud data for electric vehicles. Applied Energy 2022;317:119168. https://doi.org/10.1016/j.apenergy.2022.119168.
[22] Liu B, Jia Y, Li J, Yin S, Yuan C, Hu Z, et al. Safety issues caused by internal short circuits in lithium-ion batteries. J Mater Chem A 2018;6:21475–84. https://doi.org/10.1039/C8TA08997C.
[23] Qiao D, Wei X, Jiang B, Fan W, Gong H, Lai X, et al. Data-Driven Fault Diagnosis of Internal Short Circuit for Series-Connected Battery Packs Using Partial Voltage Curves. IEEE Trans Ind Inf 2024;20:6751–61. https://doi.org/10.1109/TII.2024.3353872.
[24] Zhao Y, Wang Z, Sun Z, Liu P, Cui D, Deng J. Data-Driven Lithium-Ion Battery Degradation Evaluation Under Overcharge Cycling Conditions. IEEE Trans Power Electron 2023;38:10138–50. https://doi.org/10.1109/TPEL.2023.3280576.
[25] Zhou X, Wang Z, Sun B, Zhang W, Zhang C, Huang Q, et al. Study of lithium-ion battery module external short circuit risk and protection design. Journal of Energy Storage 2024;86:111070. https://doi.org/10.1016/j.est.2024.111070.
[26] Zheng Y, Luo Q, Cui Y, Dai H, Han X, Feng X. Fault identification and quantitative diagnosis method for series-connected lithium-ion battery packs based on capacity estimation. IEEE Transactions on Industrial Electronics 2021;69:3059–67.
[27] Wang Q, Wang Z, Zhang L, Liu P, Zhou L. A Battery Capacity Estimation Framework Combining Hybrid Deep Neural Network and Regional Capacity Calculation Based on Real-World Operating Data. IEEE Trans Ind Electron 2023;70:8499–508. https://doi.org/10.1109/TIE.2022.3229350.
[28] Schomburg F, Heidrich B, Wennemar S, Drees R, Roth T, Kurrat M, et al. Lithium-ion battery cell formation: status and future directions towards a knowledge-based process design. Energy Environ Sci 2024:10.1039.D3EE03559J. https://doi.org/10.1039/D3EE03559J.
[29] Sun Z, Han Y, Wang Z, Chen Y, Liu P, Qin Z, et al. Detection of voltage fault in the battery system of electric vehicles using statistical analysis. Applied Energy 2022;307:118172. https://doi.org/10.1016/j.apenergy.2021.118172.
[30] Zhang J, Wang Y, Jiang B, He H, Huang S, Wang C, et al. Realistic fault detection of li-ion battery via dynamical deep learning. Nat Commun 2023;14:5940. https://doi.org/10.1038/s41467-023-41226-5.

[31] Lu J, Xiong R, Tian J, Wang C, Sun F. Deep learning to predict battery voltage behavior after uncertain cycling-induced degradation. Journal of Power Sources 2023;581:233473. https://doi.org/10.1016/j.jpowsour.2023.233473.
[32] Zhao J, Ling H, Liu J, Wang J, Burke AF, Lian Y. Machine learning for predicting battery capacity for electric vehicles. eTransportation 2023;15:100214. https://doi.org/10.1016/j.etran.2022.100214.
[33] Cui B, Wang H, Li R, Xiang L, Du J, Zhao H, et al. Internal short circuit early detection of lithium-ion batteries from impedance spectroscopy using deep learning. Journal of Power Sources 2023;563:232824. https://doi.org/10.1016/j.jpowsour.2023.232824.
[34] Cui B, Wang H, Li R, Xiang L, Du J, Zhao H, et al. Long-sequence voltage series forecasting for internal short circuit early detection of lithium-ion batteries. Patterns 2023;4:100732. https://doi.org/10.1016/j.patter.2023.100732.
[35] Wang Q, Wang Z, Liu P, Zhang L, Sauer DU, Li W. Large-scale field data-based battery aging prediction driven by statistical features and machine learning. Cell Reports Physical Science 2023;4:101720. https://doi.org/10.1016/j.xcrp.2023.101720.
[36] Chang C, Zhou X, Jiang J, Gao Y, Jiang Y, Wu T. Electric vehicle battery pack micro-short circuit fault diagnosis based on charging voltage ranking evolution. Journal of Power Sources 2022;542:231733. https://doi.org/10.1016/j.jpowsour.2022.231733.
[37] Xia B, Shang Y, Nguyen T, Mi C. A correlation based fault detection method for short circuits in battery packs. Journal of Power Sources 2017;337:1–10. https://doi.org/10.1016/j.jpowsour.2016.11.007.
[38] Kang Y, Duan B, Zhou Z, Shang Y, Zhang C. Online multi-fault detection and diagnosis for battery packs in electric vehicles. Applied Energy 2020;259:114170. https://doi.org/10.1016/j.apenergy.2019.114170.
[39] Zhang K, Hu X, Liu Y, Lin X, Liu W. Multi-fault detection and isolation for lithium-ion battery systems. IEEE Transactions on Power Electronics 2021;37:971–89.
[40] Kang Y, Duan B, Zhou Z, Shang Y, Zhang C. A multi-fault diagnostic method based on an interleaved voltage measurement topology for series connected battery packs. Journal of Power Sources 2019;417:132–44. https://doi.org/10.1016/j.jpowsour.2019.01.058.
[41] Gu X, Shang Y, Kang Y, Li J, Mao Z, Zhang C. An Early Minor-Fault Diagnosis Method for Lithium-Ion Battery Packs Based on Unsupervised Learning. IEEE/CAA J Autom Sinica 2023;10:810–2. https://doi.org/10.1109/JAS.2023.123099.
[42] Sun Z, Wang Z, Chen Y, Liu P, Wang S, Zhang Z, et al. Modified relative entropy-based lithium-ion battery pack online short-circuit detection for electric vehicle. IEEE Transactions on Transportation Electrification 2021;8:1710–23.
[43] Shang Y, Lu G, Kang Y, Zhou Z, Duan B, Zhang C. A multi-fault diagnosis method based on modified Sample Entropy for lithium-ion battery strings. Journal of Power Sources 2020;446:227275. https://doi.org/10.1016/j.jpowsour.2019.227275.
[44] Gao W, Zheng Y, Ouyang M, Li J, Lai X, Hu X. Micro-Short-Circuit Diagnosis for Series-Connected Lithium-Ion Battery Packs Using Mean-Difference Model. IEEE Trans Ind Electron 2019;66:2132–42. https://doi.org/10.1109/TIE.2018.2838109.
[45] Cao R, Zhang Z, Lin J, Lu J, Zhang L, Xiao L, et al. Reliable Online Internal Short Circuit Diagnosis on Lithium-Ion Battery Packs via Voltage Anomaly Detection Based on the Mean-Difference Model and the Adaptive Prediction Algorithm. Batteries 2022;8:224. https://doi.org/10.3390/batteries8110224.

Supplementary

Enhancing Battery Durable Operation: Multi-Fault Diagnosis and Safety Evaluation in Series-Connected Lithium-ion Battery Systems

Note 1. Detailed Charge and Discharge Regime of Cycling Experiment

Prior to serial connection, each cell underwent a fresh cell activation and a standard capacity test. The charging and discharging of series-connected batteries adhered to the benchmark values of 4.2 V for charging and 3 V for discharging. To mimic real-world operational driving conditions, we employ multi-stage constant current (CC) and Dynamic Stress Test (DST) procedures as displayed in **Fig. S1.**

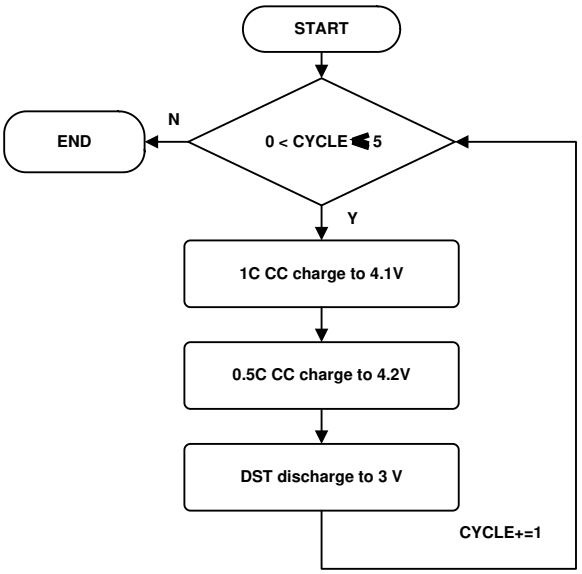

Fig. S1 The flow block diagram of the cyclic experiment.

Note 2. Predictor for Reference Voltage Responses at Cell Level

Recurrent Neural Network (RNN) is specifically designed to handle sequences of data, which is well-suited for time series analysis [1]. It can capture dependencies and patterns in the data that evolve over time. It is crucial in understanding and predicting time-dependent phenomena. RNNs can be adapted to one-step-ahead prediction, multi-step prediction and anomaly detection. It can also be powered with advanced variants like Long Short-Term Memory (LSTM) or Gated Recurrent Unit (GRU) to capture longer-range dependencies. It should be also noted that the process is a classical multi-step forecasting model. Considering the handling capacity for vanishing- and exploding gradient, and lower calculation complexity, GRU-based is used to establish the framework rather than RNN or LSTM. Here, 3-layer GRUs are organized as the core layer to model the *Predictor*, and each of them embodies

50 recurrent units (see **Fig. S2**). To enhance the robustness of the model, some Dropout layers are established between each core-layer. Finally, a Dense layer with Sigmoid activation function is used to output the forecasted battery voltage. Specifically, the input list can be described as **Formulation S1**.

$$X_{Predictor} = \{K_1, K_2, K_3, ..., K_N\} \tag{S1}$$

where, every K in the $X_{Predictor}$ represents every input of the *Predictor*; N represents the number of the time window. And K_N can be expanded a matrix as **Formulation S2**:

$$K_N = \begin{bmatrix} TV_t & TV_{t+1} & & TV_{t+500} \\ I_t & I_{t+1} & ... & I_{t+500} \\ Q_t & Q_{t+1} & & Q_{t+500} \end{bmatrix} \tag{S2}$$

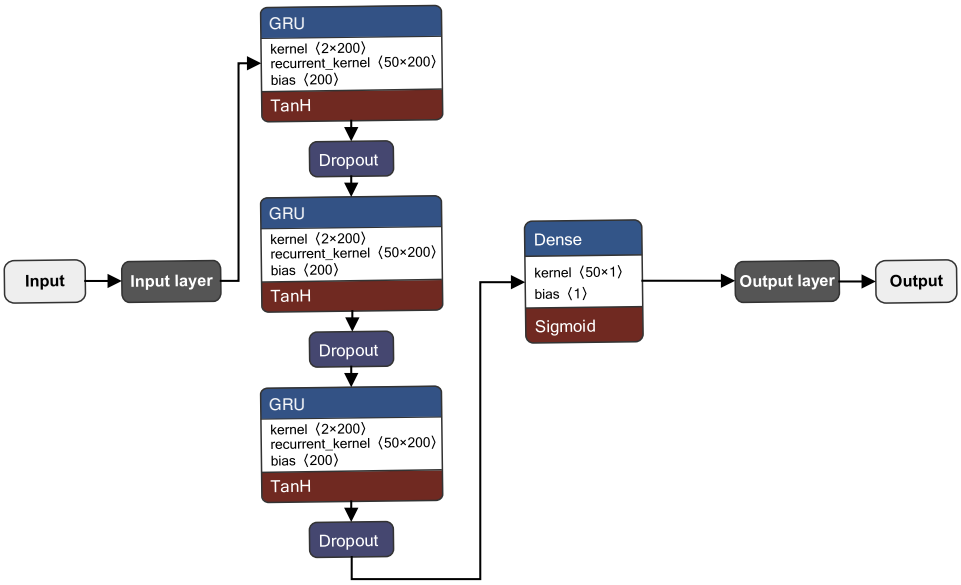

Fig. S2 Detailed algorithm structure of the *Predictor.* The total framework for the *Predictor*, including 3 GRU-based layers, 3 Dropout layers and 1 Dense-based layer (generated by Netron). The activation functions are described with a deep red box. This framework was established using ***Keras version 2.10.0.***

Note 3. *Classifier for Battery States at Module Level*

Convolutional neural network (CNN) is recognized for its success in image recognition tasks, while it can also be applied to time series classification tasks [2]. The key idea is to leverage convolutional operations to automatically learn hierarchical representations from the temporal data. CNN excels in time series classification due to their ability to automatically extract local features, handle translational invariance, and share parameters, reducing computational complexity. They capture multi-scale features and manage multi-channel data effectively, enhancing model performance. Moreover, adding attention mechanisms to CNN for time series classification enhances feature selection by focusing on important time steps or features, improves model interpretability by highlighting critical parts of the sequence, captures long-term dependencies, dynamically adjusts weights for better adaptability, and reduces information loss. This results in improved classification accuracy and a more transparent decision-making process [3]. The proposed classifier has combined CNN with a multi-head attention (CNN-mAtt) mechanism to achieve a better performance for multiple faults classification. The detailed structure of this model can be found in **Fig. S3**. Specifically.

Input layer: Considering the application of the *Predictor*, we also intercept 500-second sequence *VD* data with a 1-second stride to slide the time windows (see **Formulation S3**). The input to the Classifier is a time series sequence, where each data point represents a measurement at a specific time step.

$$X_{Classifier} = \begin{bmatrix} x_1 & x_2 & \cdots & x_{T-500} \\ x_2 & x_3 & \cdots & x_{T-499} \\ & & \cdots & \\ x_{500} & x_{501} & \cdots & x_T \end{bmatrix} \tag{S3}$$

where $X_{Classifier}$ is the matrix for the input data; each x in the matrix represents the *VD* of each timeslot; T is the timeslot of the time series.

Convolutional layers: we utilize 2 layers of 1D CNN, the first of which consists of 256 filters with a kernel size of 10, while the second involves 128 filters with the same kernel size as the first layer. In addition, a Max pooling with a pool size of 3 is utilized to reduce the spatial dimensionality and further extract hierarchical features from the input. The convolutional operation is defined as **Formulation S4**:

$$C_i = f(W_i X_{Classifier} + b_i) \tag{S4}$$

where b_i is the bias term of the i-th filter; W_i is the weight matrix of the i-th filter; C_i is the output feature map of the filter matrix and the $f(\cdot)$ is an activation function (In this paper, we select '*ReLu*' for its adaption to time series tasks).

Fully-connected layers: we set two dense layers with 128 neurons for further refining the learned representations.

Attention layer: A Multi-Head Attention layer with 5 attention heads and a 50-dimensionality key vector is applied here in our model. For the multi-head attention mechanism, multiple heads perform in parallel to pursue a better performance of the model, and the results are concentrated a linearly transformed. The attention mechanism can be mathematically represented as follows for a single head as shown in **Formulation S5**:

$$\text{Attention}(Q,K,V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \quad \text{(S5)}$$

where Q,K,V are the query, key, and value matrices; d_k is the dimensionality of the key vectors; softmax(.) represents the normalization function to obtain the attention weights.

Output layer: A three-category labeled data representing the classifications of batteries are outputted, including F-F, SC and CA, respectively.

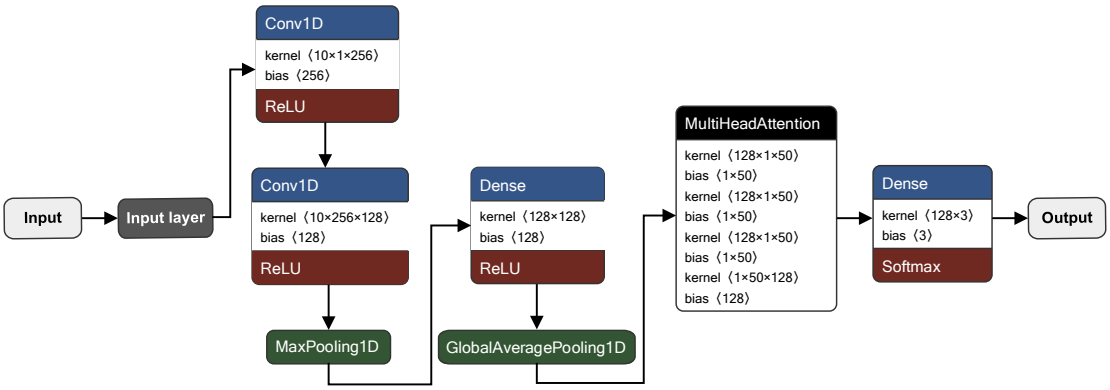

Fig. S3 Detailed algorithm structure of the *Classifier*. The total framework for the *Classifier*, including 2 Conv1D-based layers, 1 MaxPooling1D layer, 1 GlobalAveragePolling1D layer, a Multi-Head Attention layer and 2 Dense-based layers (generated by *Netron*). The activation functions are shown in deep red boxes and the parameters are provided in each module.

Table. S1 GRUs of different structures for comparison

	Layer No.	Units
Proposed	#1	100
	#2	100
	#3	100
GRU1	#1	256
	#2	256
	#3	256
GRU2	#1	100
	#2	100
	#3	100
	#4	100

Note 4. ***Data Preprocessing for Classifier Training***

There are two main steps for the data pre-processing for Classifier training, including One-hot Encoding Method and Random Shuffling (RS). Specifically,

Step1: One-hot encoding is a crucial procedure in machine learning and deep learning, involving the conversion of categorical data variables [4] . This transformation enhances the predictive capabilities and classification accuracy of a model when presented to machine learning and deep learning algorithms. It serves as a prevalent method for preprocessing categorical features in machine learning models. In this paper, we transformed all categorical labels into one-hot encoded matrices (see the first step in **Fig. S4**).

Step2: We regard RS as an indispensable phase in our classifier training process, and aim to elucidate its significance here. RS constitutes a conventional practice in DL classification pipelines, involving the random reordering of examples within datasets before the commencement of DL model training [5]. Given the specialized sorting of our battery data, organized cell by cell and cycle by cycle (where all instances of one class follow another in a specific order), there exists a risk that the model may glean patterns influenced more by the order than the intrinsic characteristics of the data. RS proves invaluable in mitigating such concerns, ensuring that the model remains impervious to order-induced patterns.

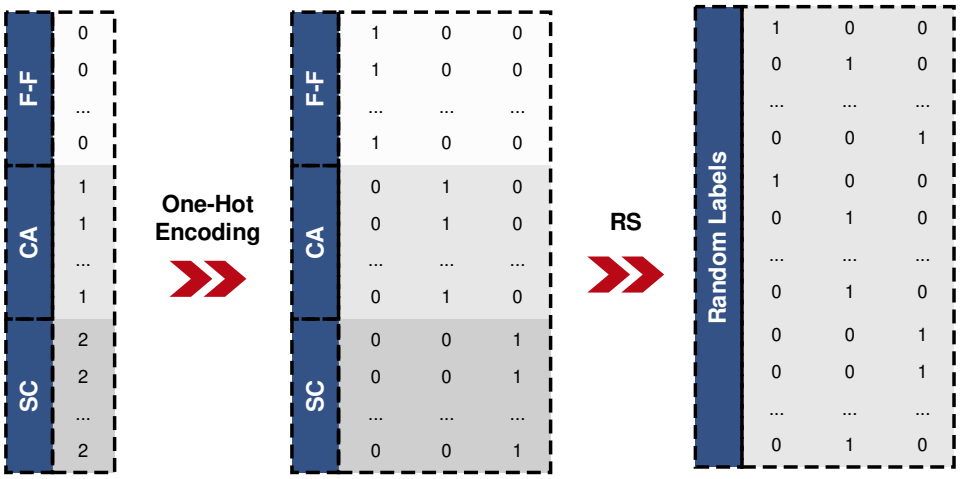

Fig. S4 The processing of one-hot encoding and RS of input data for Classifier. There are three labels for the battery data. Therefore, after the process of the One-Hot Encoding, F-F, CA, SC will be transferred into [1,0,0], [0,1,0] and [0,0,1], respectively. Origin label sequence was also be disarrayed using RS method. Above processes are coded using *Numpy* in

Python library.

Note 5. ***Model Evaluation for Predictor***

In this work, some classical indicators are used to assess the performance of the Predictor. Mean Squared Error (MSE) is a common metric to measure the accuracy of a prediction model. It represents the average of the squared differences between the predicted value and the true value (shown in formulation (S6)). Mean Absolute Error (MAE) is always used to represent the average level of the absolute error (shown in formulation (S7)). Mean Absolute Percentage Error (MAPE) is always used to evaluate the performance of the regression model. The closer the value is to 0, the smaller the difference between the predicted value and the true value (shown in formulation (S8)). In addition, we also use R^2 to evaluate the model performance, which is provided in formulation (S9). These various metrics can provide a relatively comprehensive assessment to show a convincing prediction model.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2 \tag{S6}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right) \tag{S7}$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{y_i - \hat{y}_i}{y_i}\right)\times 100\% \tag{S8}$$

$$R^2 = 1 - \frac{\sum_i\left(\hat{y}_i - y_i\right)^2}{\sum_i\left(\bar{y} - y_i\right)^2} \tag{S9}$$

where N is the total sample used in the model with the index n; y_i is the ground truth; $\hat{y}_i$ is the predicted value.

Note 6. ***Model Evaluation for Classifier***

We have selected three important indicators to evaluate the performance of the ***Classifier***, including Confusion Matrix, Receiving Operating Characteristics (ROC) Curve and Area Under ROC Curve (AUC). Specifically,

Indicator 1: The confusion matrix plays a vital role in evaluating the classification performance of CNN. It provides a detailed classification of the prediction results in different categories, which allows a thorough analysis of its strengths and weaknesses (see **Fig. S5**). Each column of the confusion matrix represents the predicted category, and the total number of the data in each column indicates the number of data predicted to be in that category; Each row represents the true category to which the data belongs, and the total number of data in each row represents the number of instances of data in that category; the value in each column represents the number of real data predicted to be in that category. In addition, accuracy (Acc) is used to evaluate the overall performance of the classifiers, which demonstrates the ratio of correctly classified samples to the total number of samples (see Formulation S10).

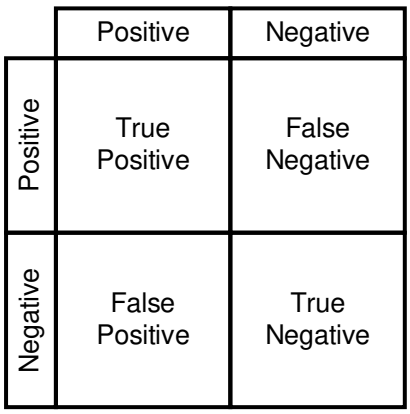

Fig. S5 Schematic diagram of confusion matrix [6]. True Positive: true class. The true class of the sample is positive and the result recognized by the model is also positive. False Negative: false negative class. The true class of the sample is a positive class, but the model recognizes it as a negative class. False Positive: false positive category. The true category of the sample is negative, but the model recognizes it as positive. True Negative: The true category of the sample is negative and the model recognizes it as negative.

Indicator 2: The ROC curve is generated by graphing the True Positive Rate (TPR) against the False Positive Rate (FPR) across different threshold settings. Each point on the ROC curve signifies a distinct trade-off between sensitivity and specificity. The diagonal line in the ROC space symbolizes random guessing, with points positioned above this line

denoting performance superior to random chance (see **Fig. S6**) [7].

Indicator 3: AUC generally measures the area under ROC curve and quantify the overall performance of the classification model across different threshold settings [8]. A higher AUC value indicates a better balance between TPR and FPR, suggesting superior classification performance.

Because the recognition process is a multi-classification model, the model needs to transfer the classification label into a binarization matrix in order to evaluate the model performance. Micro average ROC and Macro average ROC are both provided to evaluate the total performance of the model comprehensively. Primarily, based on the confusion matrix, TPR and FPR can be utilized to generate the ROC curve and calculate the AUC. The calculation methods are provided as the following formulations:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{S10}$$

$$\text{macro } TPR = \frac{1}{3}\sum_{c=1}^{3}\frac{TP_c}{TP_c + FN_c} \tag{S11}$$

$$\text{macro } FPR = \frac{1}{3}\sum_{c=1}^{3}\frac{FP_c}{TN_c + FP_c} \tag{S12}$$

$$\text{micro } TPR = \frac{\sum_{c=1}^{3} TP_c}{\sum_{c=1}^{3} TP_c + \sum_{c=1}^{3} FN_c} \tag{S13}$$

$$\text{micro } FPR = \frac{\sum_{c=1}^{3} FP_c}{\sum_{c=1}^{3} TN_c + \sum_{c=1}^{3} FP_c} \tag{S14}$$

where TP_c is the True Positive (TP) samples that the model predicts are positive of c^{th} clustering class; FP_i is the Fale Positive (FP) samples that the model predicts are negative of c^{th} clustering class; TN_c is the True Negative (TN) samples that the model predicts are negative of c^{th} clustering class; FN_c is the False Negative samples that the model predicts are negative of c^{th} clustering class.

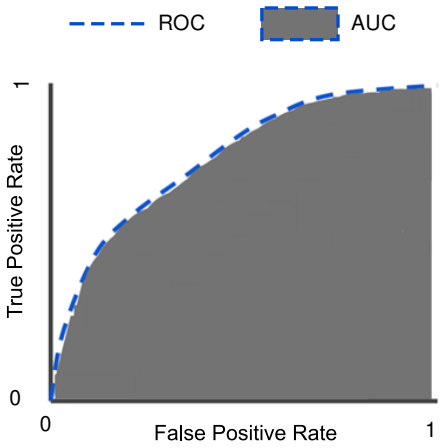

Fig. S6 Schematic diagram of RUC and AUC [8] . The dashed blue line represents the ROC curve; the grey areas represent the AUC. When the AUC is closer to 0.5 means the model is less effective in classification, and closer to 1 means the classification is more effective. When the area is between [0.5,0.7] indicating that the model is generally better at classification than random classification; when the area is between [0.7,0.85], it indicates that the model has a good classification; when the area is between [0.85,0.95] it indicates that the model has an excellent classification.

Note S7. ***Further comparison between the proposed method and non-AI methods***

This paper primarily employs deep learning methods to construct a multi-fault diagnosis framework for batteries. In the main text, we compare the performance of similar machine learning algorithms. However, some non-AI-driven algorithms are also widely used in battery fault diagnosis. Here, we focus on comparing the proposed method with several typical non-AI methods in terms of computation time and diagnostic effectiveness. As shown in the **Table. S2** below, our proposed method leads non-AI based methods in both computational efficiency and diagnostic time, further demonstrating the advanced nature of the proposed approach. Our diagnostic framework can detect faults during the initial stages, specifically within the three cycles preceding TR. In contrast, non-AI methods can only diagnose faults close to the onset of thermal runaway, which results in insufficient time for accident prevention in real-world applications. Additionally, most non-AI methods can only detect anomalies in individual parameters and struggle to identify multiple faults simultaneously.

Table. S2 Comparison between proposed method and non-AI based methods

Method	Computing Time (s)	Detection Time (before TR)	Fault Type
Proposed	**3.82**	**three cycles earlier**	**Yes**

PCC	5.73	last cycle	No
Shannon Entropy	7.11	last cycle	No
Mean-Difference Model	15.96	last cycle	No

References of the Supplementary

[1] Recurrent Neural Network - an overview | ScienceDirect Topics n.d. https://www.sciencedirect.com/topics/engineering/recurrent-neural-network (accessed November 13, 2023).

[2] Hatami N, Gavet Y, Debayle J. Classification of Time-Series Images Using Deep Convolutional Neural Networks 2017. https://doi.org/10.48550/arXiv.1710.00886.

[3] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention Is All You Need 2023. https://doi.org/10.48550/arXiv.1706.03762.

[4] Okada S, Ohzeki M, Taguchi S. Efficient partition of integer optimization problems with one-hot encoding. Sci Rep 2019;9:13036. https://doi.org/10.1038/s41598-019-49539-6.

[5] Lesson 1.4 Data Shuffling. Deci n.d. https://deci.ai/course/data-shuffling/ (accessed November 13, 2023).

[6] Classification: ROC Curve and AUC | Machine Learning. Google for Developers n.d. https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc (accessed November 27, 2023).

[7] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 2011;12:2825–30.

[8] Cui D, Wang Z, Liu P, Wang S, Zhao Y, Zhan W. Stacking regression technology with event profile for electric vehicle fast charging behavior prediction. Applied Energy 2023;336:120798. https://doi.org/10.1016/j.apenergy.2023.120798.