Energy 292 (2024) 130584

Contents lists available at ScienceDirect

Energy

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

A novel hybrid deep learning model for accurate state of charge estimation of Li-Ion batteries for electric vehicles under high and low temperature

Muhammad Hamza Zafar [a], Noman Mujeeb Khan [b], Mohamad Abou Houran [c], Majad Mansoor [d,f], Naureen Akhtar [a], Filippo Sanfilippo [a,e,*]

[a] Department of Engineering Sciences, University of Agder, Grimstad, 4879, Norway
[b] Department of Electrical Engineering, Capital University of Science and Technology, Islamabad, 44000, Pakistan
[c] School of Electrical Engineering, Xi'an Jiaotong University, Xi'an, 710049, China
[d] Department of Automation, University of Science and Technology of China, Hefie, 230026, China
[e] Department of Software Engineering, Kaunas University of Technology, 44029 Kaunas, Lithuania
[f] Ningbo China Institute for Supply Chain Innovation (NISCI), Ningbo, 315000, China

ARTICLE INFO

Dataset link: https://calce.umd.edu/battery-data#Storage

Keywords:
State of charge
Electric vehicles
Deep learning
Evolutionary intelligence
High and low temperatures

ABSTRACT

This paper presents a novel architecture, termed Fusion-Fission Optimisation (FuFi) based Convolutional Neural Network with Bi-Long Short Term Memory Network (FuFi-CNN-Bi-LSTM), to enhance state of charge (SoC) estimation performance. The proposed FuFi-CNN-Bi-LSTM model leverages the power of both Convolutional Neural Networks (CNN) and Bi-Long Short Term Memory Networks (Bi-LSTM) while utilizing FuFi optimization to effectively tune the hyperparameters of the network. This optimization technique facilitates efficient SoC estimation by finding the optimal configuration of the model. A comparative analysis is conducted against FuFi Algorithm-based models, including FuFi-CNN-LSTM, FuFi-Bi-LSTM, FuFi-LSTM, and FuFi-CNN. The comparison involves assessing performance on SoC estimation tasks and identifying the strengths and limitations of models. Furthermore, the proposed FuFi-CNN-Bi-LSTM model undergoes rigorous testing on various drive cycle tests, including HPPC, HWFET, UDDS, and US06, at different temperatures ranging from -20 to 25 degrees Celsius. The model's robustness and reliability are assessed under different real-world operating conditions using well-established evaluation indexes, including Relative Error (RE),Mean Absolute Error (MAE), R Square (R^2), and Granger Causality Test. The results demonstrate that the proposed FuFi-CNN-Bi-LSTM model achieves efficient SoC estimation performance across a wide range of temperatures at higher and lower ranges. This finding signifies the model's efficacy in accurately estimating SoC in various operating conditions.

1. Introduction

The storage of renewable and other forms of energy is encouraged to be done via rechargeable batteries [1]. Lithium-ion (Li-Ion) batteries having the benefits of high energy density, reliable protection, and eco-friendly nature dominate the battery industry globally in Electrical Vehicles (EVs) and power saver bank applications. The most important parameter for information-dependent goals for lithium-ion batteries is the state of charge (SoC). SoC is sensitive to several long and short term parameters and is defined as the proportion of capacity remaining to the current rated capacity [2]. Making sure the battery operates within a safe operating range is advantageous for the battery management system (BMS) [3]. Hence it is an important factor to estimate distance under load and intrinsic state of charge of the battery under mechanical load. Lithium-ion batteries can function well if battery SoC is accurately estimated [4,5].

The BMS task of estimating the SOC with accuracy is complex, challenging as well as crucial for ensuring optimal performance due to following key factors:

- Non-Linear Battery Behavior: batteries exhibit non-linear behavior in the battery's voltage and its SoC. The voltage curve is influenced by various factors, such as battery chemistry, service life, and current flow, making it difficult to directly correlate voltage measurements with accurate SoC estimation.
- Battery Ageing and Degradation: over time, batteries experience capacity degradation, reducing their ability to hold a charge. This degradation is influenced by factors like charge and discharge rates, operating conditions, material degradation, and the number of charge cycles. Estimating SoC accurately requires accounting

* Corresponding author at: Department of Engineering Sciences, University of Agder, Grimstad, 4879, Norway.
E-mail address: filippo.sanfilippo@uia.no (F. Sanfilippo).

https://doi.org/10.1016/j.energy.2024.130584
Received 28 September 2023; Received in revised form 18 January 2024; Accepted 3 February 2024
Available online 7 February 2024

for the battery ageing and adjusting the estimation algorithms accordingly.

- Measurement Errors: measurement errors can occur due to the limitations of the battery management system (BMS) or the measurement equipment itself. Inaccurate voltage, current, or temperature measurements can lead to erroneous SoC estimations.
- Coupling Effects: the SoC estimation process becomes more complex when considering the coupling effects of other battery parameters. For example, temperature significantly impacts the battery's performance and influences its internal resistance and capacity. Ignoring these factors can lead to inaccurate SoC estimations.

The lithium ion batteries are expected to dominate the global electric vehicle battery market, with a market share of about 40 percent. The user satisfaction in adoption of this technology is highly dependent upon the dependable information regarding reliable range and load monitoring. Each battery chemistry and design have unique characteristics, making it challenging to develop universal models that accurately estimate SoC for all types of batteries. This information heavily relies on estimating the SoC of a battery. SoC is a challenging task due to several factors. The first and the most important factor is its nonlinear charging and discharging curves with small quasi linear regions [6]. This non-linearity is compounded with hysteresis implying non-linear relationship between the battery's voltage and its state of charge. The factors influencing deviation are temperature, ageing, and discharge rate. The hysteresis implies that the voltage response during charging is different from the voltage response during discharging. These non-linear and hysteresis effects make it difficult to accurately estimate the SoC based solely on voltage measurements. Batteries experience degradation over time, which affects the capacity to attain charge and voltage characteristics. Battery capacity decreases with age reducing the ability to hold charge [7]. Battery performance is dependent on operating temperature. Temperature variations affect the internal resistance, self-discharge rate, and capacity of the battery. Different temperature conditions require different models or algorithms to estimate the SoC accurately. Managing temperature effects and incorporating them into the estimation process adds complexity to the task. The internal battery parameters are not directly observable [8]. Since SoC estimation relies on indirect measurements and models, it can introduce additional uncertainties and errors. SoC drift occurs during Coulomb counting, which involves integrating the current flow in and out of the battery. This is prone to errors over time due to drift in the measurement of current and voltage, as well as inaccuracies in the capacity estimation caused by hysteresis. The model specific algorithms need to be developed and calibrated for different battery chemical composition and configurations adding complexity to the estimation process. The non-linear voltage characteristics, temperature effects, measurement inaccuracies, lack of universal models, limited observability, and errors associated with Coulomb counting are equally important for enhanced SoC.

Moreover, accurately estimating SoC becomes even more relevant when considering real-life operational conditions, such as low and high temperatures. Temperature variations affect the battery's internal resistance, chemical reactions, and self-discharge rates, all of which impact SoC estimation. In cold temperatures, batteries may exhibit lower capacity, leading to overestimation of SoC. Conversely, in high temperatures, the battery's capacity may increase temporarily, leading to underestimation of SoC. Therefore, accounting for temperature effects and adjusting the estimation algorithms accordingly is crucial to obtain accurate SoC estimations in real-life scenarios.

1.1. Literature review

SoC cannot be directly measured since it is an internal property of lithium-ion battery [9–11]. With traditional methods, only outside factors like voltage, current, and temperature may be used to estimate it. SoC estimate error cannot be avoided particularly caused by the sensor measurements [12]. More complexity to SoC is added at low temperatures due to the loss in capacity increases since the battery's chemical reaction rate slows down [13].

Wintertime temperature lows can reach 0 °C in the majority of the geographies [14]. When lithium-ion batteries are running at low temperatures, the estimation error of the SoC is substantially bigger. In high latitudes, wintertime low temperatures can range from 10° to 20° degrees Celsius. It is crucial to increase the precision and robustness of SoC estimates in low-temperature conditions if lithium-ion batteries are to function properly in these areas. Lithium-ion batteries may be used for a wider variety of purposes thanks to this, and it can also give batteries operating in natural settings more stability. The high-performance materials and cutting-edge manufacturing technologies such as Metal–organic frameworks, provide tremendous potential for enhancing lithium-ion batteries' energy density [15]. It comes at the cost of rapid degradation of electrolytes leading to capacity degradation in a low temperature setting. There is not enough research on lithium-ion battery SoC estimate in low-temperature settings. An analogous circuit model was developed to describe the performance of lithium-ion batteries. By using extended Kalman filtering (EKF), they calculated the SoC at below-freezing temperatures [16]. Although the inaccuracy is below 4.5%–6%, it is still not adequate for EV platforms to estimate distances based on inaccurate SoC.

Analytical models established a mathematical dependency between open circuit voltage (OCV) and SoC with respect to temperature. A modified OCV approach by Gong et al. [23] using dual adaptive extended Kalman filter based on the residual sequence is utilized. The prerequisites for practical use cannot be reached by this procedure due to the lack of temporal trends over large number of charging-discharging cycles achieving 50% error in initial SoC under FUDS dataset. To further improvise this approach, He et al. recommended using the voltage difference as a threshold for estimation performance and time [24]. These techniques can produce SoC estimations that are more precise, but their application range is short about 10 degrees Celsius. These classical ML models require a precise battery model. The rise of renewable energy added to difficulties for the fusion of battery models. Volterra integral dynamical models is applied to manage multiple cells of battery load-leveling problem. The real-time optimization of energy storage is ideal for the adaptive method. By examining the power load, Sidorov et al. [25] once more demonstrated the value of the Volterra approach for battery modeling. The detailed literature review of the deep learning based SoC estimation models is presented in Table 1.

The electrical characteristics of lithium-ion batteries are sophisticated and highly nonlinear. The forecast accuracy is restricted since the battery model does not correctly represent the battery properties. Because of its superior nonlinear fitting capabilities, the neural network can avoid the estimation errors that the model introduces. The neural network approach offers higher accuracy and redundancy. Attention-infused long short-term memory (LSTM) network by Tadele et al. attained RMSE of <1.41% at multiple magnitudes of temperatures [26]. The robustness of the strategy was confirmed by the Monte Carlo dropout methodology. The technique, however, does not take into account the SoC estimation for temperatures below zero. Bian et al. in [27] employed the bidirectional-LSTM model for SoC accuracy. LSTM adjusts the past and future directions correlations and is highly effective for learning periodic features. The authors attained <2.6% in the 0–20° range, leaving plenty of space for improvement. These techniques are challenging to use in real-world applications since temperature variance is not adequate for a wide range of operating conditions.

Table 1
Literature review of machine learning/deep learning based SoC estimation.

Ref	Year	Technique	Summary	Results
[17]	2019	ELM-GSA	This paper presents an improved ELM for estimating the SoC by estimation model. GSA is used for hyperparameter tuning and to optimize the number of neurons in a hidden layer. Hence model does not require an internal battery mathematical model. Validation experiments are conducted at different temperatures and electric vehicle drive cycles, and comparisons with other NN models are mad.	RMSE = 1.6% MAE = 1.67%
[18]	2019	Gated-RNN	A gated RCNN unit is employed for SoC from using measured current, voltage, and temperature signals. The method uses previous SoCs and measurements for better accuracy. Model well handles initial SoC magnitude and varying temperatures.	RMSE = 3.5%
[19]	2020	LSTM-UKF	A LSTM-based RNN architecture is used for to SoC from time-stamped voltage, current, and temperature variables. UKF filter is used to measured signal noise improving SoC accuracy. LSTM-UKF tackles the influence of ambient temperature well. performance evaluation is done on dynamic stress tests and driving schedules.	RMSE = 1.1% MAE = 1.0%
[20]	2020	Autoregressive GPR	A data-driven method using GPR and squared exponential kernel function is utilized. The method uses feature extraction and optimization by automatic relevance determination. The proposed method accurately approximates nonlinearity, offers nonparametric modeling, and probabilistic predictions.	RMSE = 3.21%
[3]	2022	I&I Transformer NN	A ML method using a Transformer NN and an adaptive observer is elaborated in this study that predicts SoC with current and offers richer information processing for time-dependent features by Transformers model. A higher SoC efficiency is achieved at the expense of more computation power and data preprocessing	RMSE = 0.98% MAE = 0.84%
[21]	2022	SSA-IELM	An improved ELM algorithm, using a salp swarm algorithm to propagate learning error to hidden layers. Chaotic mapping is used to make the initialized individuals uniformly distributed improvise the weights and biases of NN iteratively, and a sine cosine algorithm is embedded to improvise vectors similar to the time to vector (T2V) approaches.	MAE = 0.538 MAPE = 0.887%
[22]	2023	EI-LSTM-CO	An EI-LSTM-CO for battery SoC estimation model includes an extended input with additional slow time-varying information sliding window average with voltage and a constrained output using a state flow strategy based on the Ampere-hour integration. The experimental verification on LiFePO4 battery datasets at various temperatures is done.	RMSE = 1.3% MAXE = 3.2%

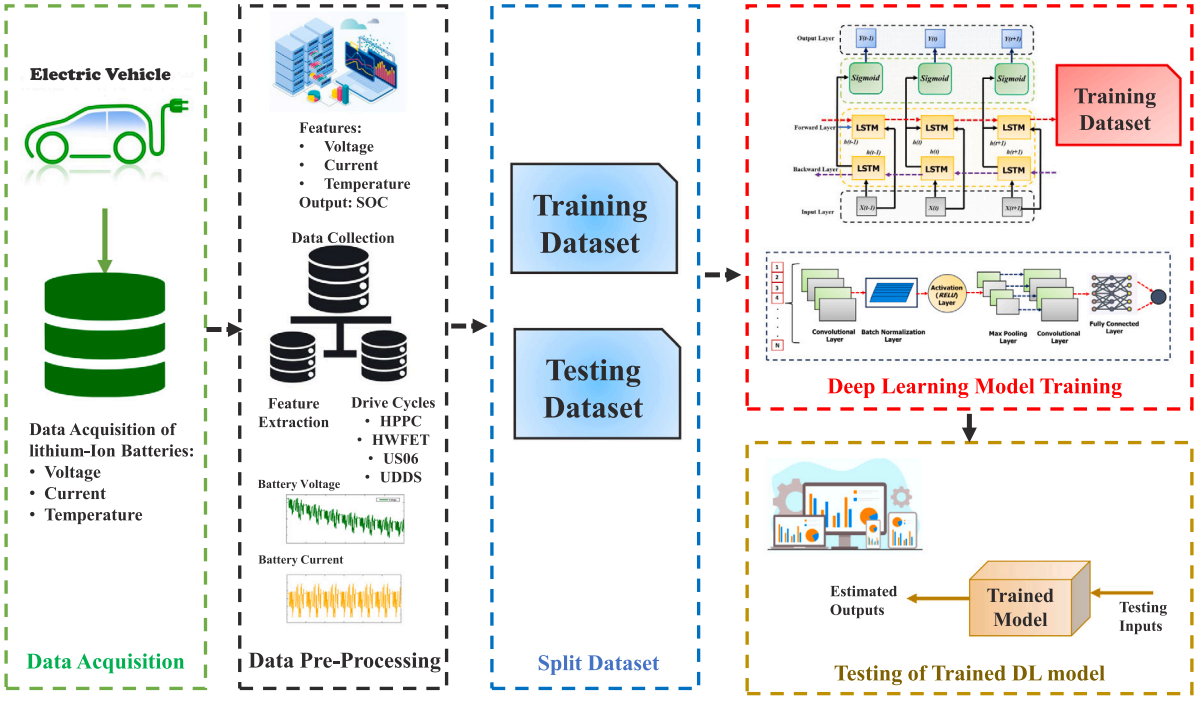

Fig. 1. Graphical abstract electric vehicles state of charge for Li-Ion batteries.

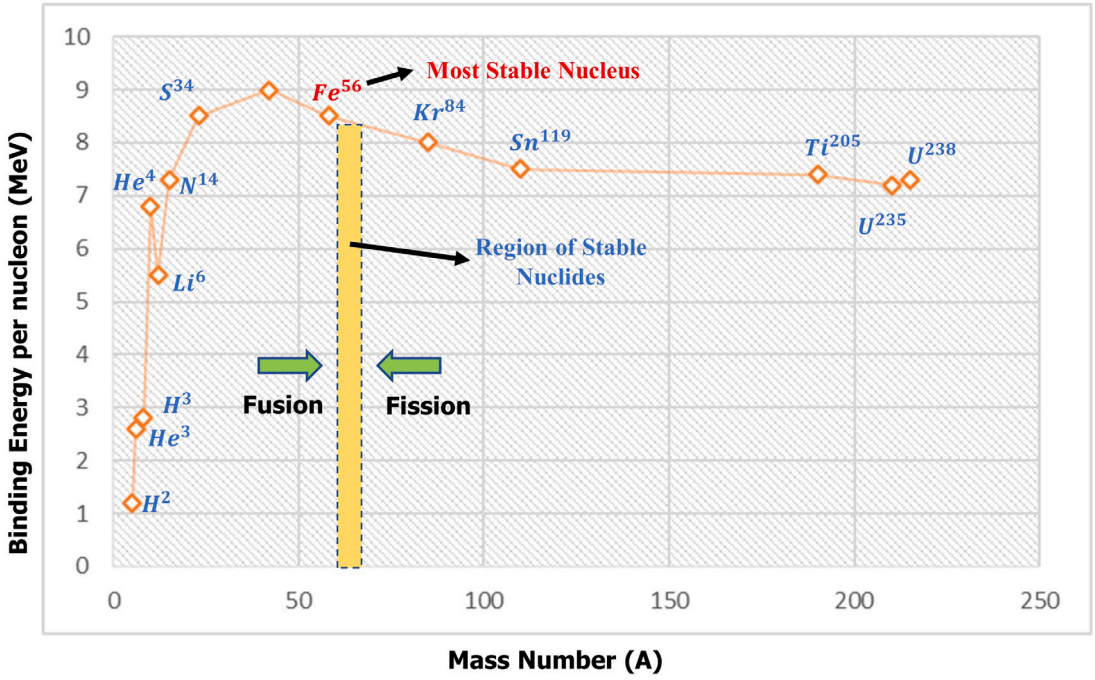

Fig. 2. Relationship between B_{avg} and mass number A.

1.2. Contributions and paper organization

In this paper, we proposed a comprehensive study that accurately predicts the SoC of Lithium-ion batteries in EVs in low and high temperatures, and the graphical abstract of the proposed model is given in Fig. 1. Contributions of this work are:

- This study conducts a thorough analysis of five unique hybrid deep learning models, focusing on their performance at both low and high temperature conditions. This analysis is critical, considering the impact of temperature variations on battery performance and the accuracy of State of Charge (SoC) estimations.
- The research introduces a novel architecture, termed Fusion-Fission Optimisation (FuFI) based CNN with Bi-LSTM (FuFi-CNN-Bi-LSTM), for SoC estimation. This architecture ingeniously combines Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory (Bi-LSTM) layers, indicating a sophisticated approach to capturing both spatial and temporal data features for enhanced SoC prediction.
- A significant contribution of this work is the efficient tuning of hyperparameters in the FuFi-CNN-Bi-LSTM model. This optimization process plays a vital role in refining the model's performance, ensuring high accuracy in SoC estimation across various conditions.
- The study includes a comprehensive comparison of the FuFi-CNN-Bi-LSTM model with other models utilizing the FuFi algorithm, namely FuFi-CNN-LSTM, FuFi-Bi-LSTM, FuFi-LSTM, and FuFi-CNN. This comparison is instrumental in demonstrating the superior performance and efficiency of the proposed model.
- The proposed model undergoes rigorous testing under diverse temperature conditions (0°, 10°, 25°, −10°, and −20 °C) and across various Drive Cycle Tests (HPPC, HWFET, UDDS, and US06). This extensive testing framework ensures the model's robustness and reliability in different environmental settings and driving scenarios.
- The research highlights the model's capability to accurately estimate SoC in batteries under extreme temperature conditions. This aspect is particularly noteworthy as accurately predicting SoC in such conditions is challenging and crucial for battery management systems.

This paper is organized into five sections, starting with an introduction in Section 1. Section 2 presents the proposed technique in detail, which aims to improve the efficiency of the SoC estimation. Section 3 provides an explanation of the datasets of drive cycles used in this study, including their sources and characteristics. Section 4 explains the results obtained from the proposed technique and provides a comprehensive discussion of the findings. This section also includes a comparison of the results with those obtained using other techniques. Finally, Section 5 presents the conclusion of the study and summarizes the main findings.

2. Proposed methodology

2.1. Fusion-Fission optimisation algorithm (FuFi)

2.1.1. Inspiration

The processes inspired by nature form the best mathematical optimizers since they evolve over million of years. On similar natural process is the binding energy of nucleous which refers to the least amount of energy required to disassemble the nucleus of an atom into its constituent nucleons, which comprise protons (Z) and neutrons (N). This binding energy is generated by the strong nuclear force, which has a positive value and attracts the nucleons towards one another. Greater binding energy contributes to increased stability of the nucleus. However, it is worth noting that the Coulomb repulsive force exhibited by the protons reduces the nuclear attraction force, leading to a decrease in binding energy. As a result, the stability of the nucleus is further diminished when more neutrons replace protons. Additionally, in nuclei, paired protons are often located in close proximity to one another, which reduces the strength of the strong nuclear force due to their repulsive force, thus resulting in instability. The detailed mathematical model of FuFi algorithm is elaborated below (see Figs. 2–4).

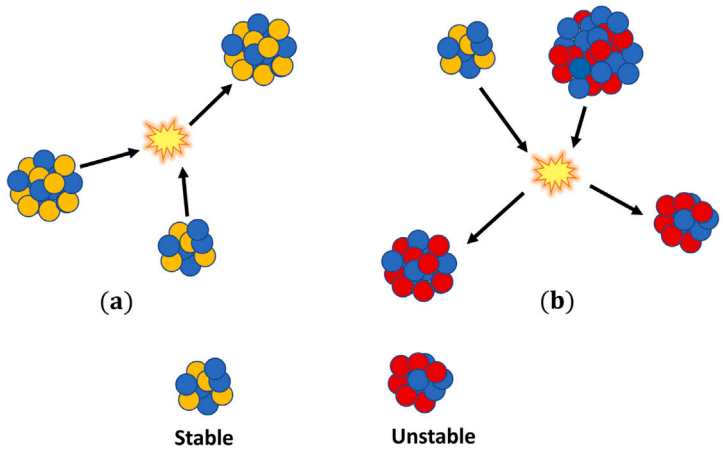

Fig. 3. Processes of nuclear fusion and fission reaction.

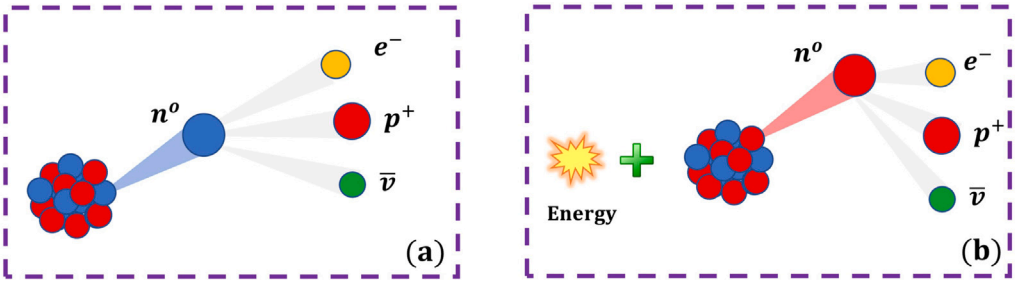

Fig. 4. Schematic illustrations of β^- and β^+ decay mechanisms.

2.1.2. Mathematical model

The mathematical model of the FuFiO algorithm [28], aims to enhance the stability of nuclei by increasing their binding energy through nuclear reactions such as fusion, fission, and B-decay. The FuFiO algorithm directs its movement towards enhancing the binding energy of nuclei. FuFiO is a population-based metaheuristic algorithm that employs a group of nuclei as the agents of the population.

The initial value of x_i^j is randomly determined using the following approach:

$$x_i^j(0) = lb^j + r(ub^j - lb^j) \tag{1}$$

where ub^j and lb^j are max and min magnitudes of search space for the jth number of dimensions with a random r in range [0,1]. These initial positions for all nuclei are then collected in the matrix $X(0)$, which represents the starting position of the population. The different types of nuclear reactions, including fusion, fission, and β-decay, are treated differently depending on the types of nuclei involved. Therefore, as depicted in Fig. 5, each group of nuclei can be updated by considering three distinct types of reactions.

Group 1: Stable nucleus

If the ith nucleus is stable (X_i^{stable}), one of the following three reactions is selected randomly:

Reaction 1:

The ith nucleus collides with another stable nucleus, and the new position is calculated using the following formula:

$$X_i^{new} = rX_i^{stable} + (1-r)X_j^{stable} \tag{2}$$

In this fusion simulation, a random vector r within the range of 0 to 1 is combined with the selection of a stable nucleus X_j^{stable}, which is random stable nuclei. This process models the collision of two stable nuclei, which generates a new nucleus. Fig. 5 visually represents the process within the reaction space by the combination of r and $1-r$.

Reaction 2:

If the ith nucleus collides with an unstable nucleus, the resulting collision produces a novel solution given by:

$$X_i^{new} = X_i^{stable} + r(X_i^{stable} - X_j^{unstable}) \tag{3}$$

The selection of an unstable nucleus $X_j^{unstable}$, chosen at random from among other unstable nuclei, is central to this reaction. The reaction, depicted in Fig. 6, models the process of fission, which occurs when an unstable nucleus collides with a stable one.

The process of β-decay in a stable nucleus is modeled by this reaction, which involves the use of p to denote a random subset of problem variables, set of all variables is d. The counter of variables is represented by k, and R denotes a random nucleus, The reaction is illustrated in Fig. 7.

Group 2: Unstable nucleus

The second group involves updating the ith nucleus by randomly selecting one of three reactions if it is unstable ($X_i^{unstable}$). The three possible reactions are:

Reaction 1:

If two unstable nuclei collide, the resulting new position is determined by the following method:

$$X_i^{new} = rX_i^{unstable} + (1-r)(X_j^{unstable} - X_i^{unstable}) \tag{4}$$

The selection of a random vector r within the interval of [0,1] and an unstable nucleus $X_j^{unstable}$, randomly selected from other unstable nuclei, is utilized in this reaction. This reaction models the process of fission, where an unstable nucleus is struck by another unstable nucleus, as illustrated in Fig. 5.

Reaction 2: If the unstable nucleus, $X_i^{unstable}$, interacts with a stable nucleus, the new position is as follows:

$$X_i^{new} = X_i^{unstable} + r(X_i^{stable} - X_j^{stable}) \tag{5}$$

where X_j^{stable} is a randomly selected stable nucleus illustrated in Fig. 9.

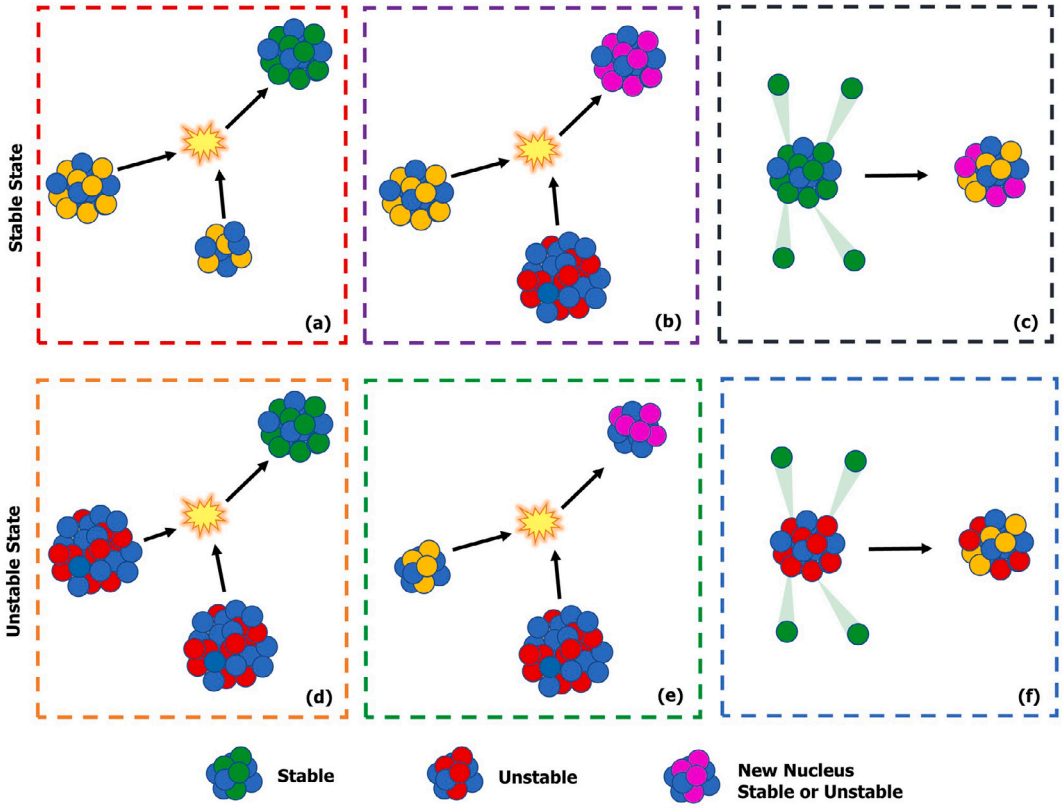

Fig. 5. Distinct reaction types to update nuclei.

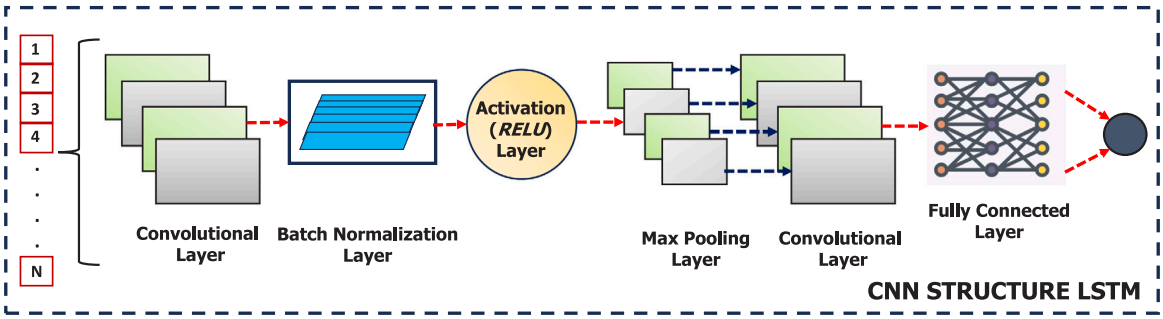

Fig. 6. General structure of CNN.

2.2. Convolutional Neural Network (CNN)

The CNN is a type of deep NN commonly used in image and video recognition [29]. It is composed of multiple layers, each of which applies a set of filters to the input data to extract features. The input data to a CNN is typically a multidimensional array, such as an image represented as a matrix of pixel values. The filters applied by each layer of the network are also multidimensional arrays, known as kernels or weights, that are learned during training. The general structure of CNN is shown in Fig. 6.

A 1D CNN is a variant of CNN used to process one-dimensional data, such as time series signals [30]. It applies a convolution operation to the input data along one dimension, with a sliding window of fixed size, to extract features from the signal. The input data to a 1DCNN is typically a sequence of real values, represented as a one-dimensional array. The filters applied by each layer of the network are also one-dimensional arrays, known as kernels or weights, that are learned during training. The output of a convolutional layer in a 1DCNN is obtained by applying a convolution operation between the input and the kernel, along the time dimension. The convolution operation is defined as:

$$(f * g)(n) = \sum_{m=-\infty}^{\infty} f(m)g(n-m), \tag{6}$$

where f and g are two functions and * denotes the convolution operator. In the case of 1DCNNs, the input is the one-dimensional array and the kernel is the filter applied to it.

The output of the convolution operation is then passed through a nonlinear activation function, such as the Rectified Linear Unit (ReLU), defined as:

$$ReLU(x) = \max(0, x) \tag{7}$$

The ReLU function is commonly used in 1DCNNs because it introduces nonlinearity and helps to avoid the vanishing gradient problem [31]. The mathematical equations that describe a convolutional layer of a 1DCNN can be represented as:

$$z_i = \sum_{j=1}^{K} x_{i+j-1} w_j + b, \tag{8}$$

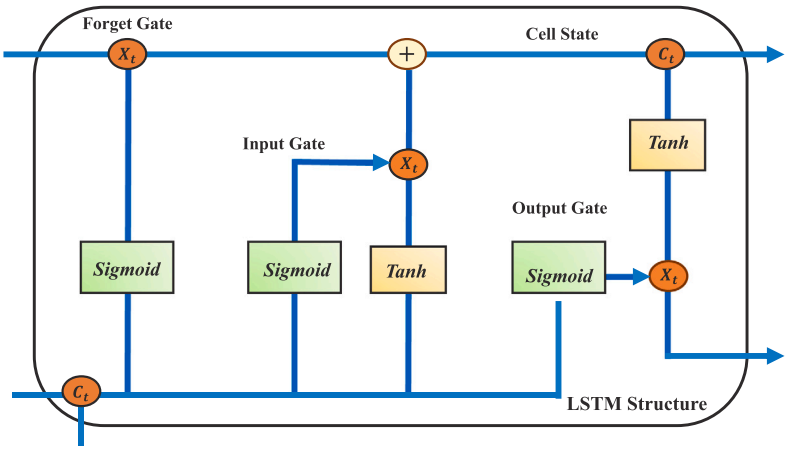

Fig. 7. General structure of LSTM cell.

where x_i is the input data at position i, w_j is the weight at position j for the ith filter, b is the bias for the ith filter, and K is the size of the kernel.

The output of the convolutional layer after the ReLU activation function can be represented as:

$$y_i = \max(0, z_i) \tag{9}$$

Finally, the output of the max-pooling layer can be represented as:

$$p_i = \max_{j=1}^{S} y_{i+(j-1)S}, \tag{10}$$

where S is the stride of the pooling operation. 1DCNNs have several advantages for regression tasks, particularly when the input data is in the form of time series. Firstly, they can capture temporal dependencies in the data by applying convolution and pooling operations along the time dimension, which can lead to improved performance compared to traditional regression models that ignore temporal information. Additionally, the use of ReLU activation functions introduces nonlinearity and helps to avoid the vanishing gradient problem, which can be a common issue in traditional neural networks. Furthermore, the ability to learn local patterns and features through convolutional layers can be advantageous in scenarios where there are complex interactions between variables.

2.3. Long Short-Term Memory Network (LSTM)

LSTM networks are a type of recurrent neural network (RNN) that are commonly used for processing sequential data, such as time series signals or natural language [32]. They are designed to address the vanishing gradient problem in traditional RNNs, which can occur when the gradient of the error signal becomes too small and the network is unable to learn long-term dependencies in the data.

The key component of an LSTM network is the memory cell, which allows the network to selectively forget or remember information from previous time steps. The memory cell is composed of several gates, which control the flow of information into and out of the cell. The gates are implemented as sigmoid or hyperbolic tangent activation functions, which can take values between 0 and 1, and −1 and 1, respectively [33]. The general structure of the LSTM cell is shown in Fig. 7

The mathematical equations that describe the LSTM network are as follows. Let x_t be the input at time step t, h_t be the hidden state at time step t, c_t be the cell state at time step t, and σ and f be the sigmoid and hyperbolic tangent functions, respectively. The LSTM equations can be written as:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i), \tag{11}$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f), \tag{12}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o), \tag{13}$$

$$\tilde{c}t = f(W_c x_t + U_c ht - 1 + b_c), \tag{14}$$

$$c_t = i_t \cdot \tilde{c}t + f_t \cdot ct - 1, \tag{15}$$

$$h_t = o_t \cdot f(c_t), \tag{16}$$

where i_t, f_t, and o_t are the input, forget, and output gates, respectively, and $\tilde{c}_t$ is the candidate cell state. The weights W_i, W_f, W_o, and W_c, and the biases b_i, b_f, b_o, and b_c, are learned during training. The function $f(c_t)$ can be any nonlinear activation function, such as the hyperbolic tangent or the Rectified Linear Unit (ReLU).

LSTM networks have several advantages for time series regression tasks [34]. Firstly, they are capable of capturing long-term dependencies in the data, which can be difficult for traditional regression models to achieve. This is achieved by selectively retaining or forgetting information from previous time steps using the memory cell and its gates. Secondly, LSTM networks can handle input sequences of variable length, making them well-suited for time series data with irregular sampling intervals. Additionally, they are able to learn complex temporal patterns and relationships in the data, which can lead to improved performance compared to traditional regression models.

2.4. Bi-LSTM network

Bidirectional LSTM (Bi-LSTM) networks are an extension of the LSTM network that are commonly used for processing sequential data, such as time series signals or natural language [35]. Bi-LSTMs process temporal dependencies in both directions the input sequence in both forward direction and backward direction simultaneously, which can improve its ability to model complex temporal patterns and relationships in the data. The general structure of Bi-LSTM network is shown in Fig. 8.

The mathematical equations that describe the Bi-LSTM network are similar to those of the traditional LSTM network, but with two sets of equations: one for the forward direction and one for the backward direction [36]. Let x_t be the input at time step t, h_t^f be the hidden state in the forward direction at time step t, h_t^b be the hidden state in the backward direction at time step t, c_t^f be the cell state in the forward direction at time step t, c_t^b be the cell state in the backward direction at time step t, and σ and f be the sigmoid and hyperbolic tangent functions, respectively. The Bi-LSTM equations can be written as:

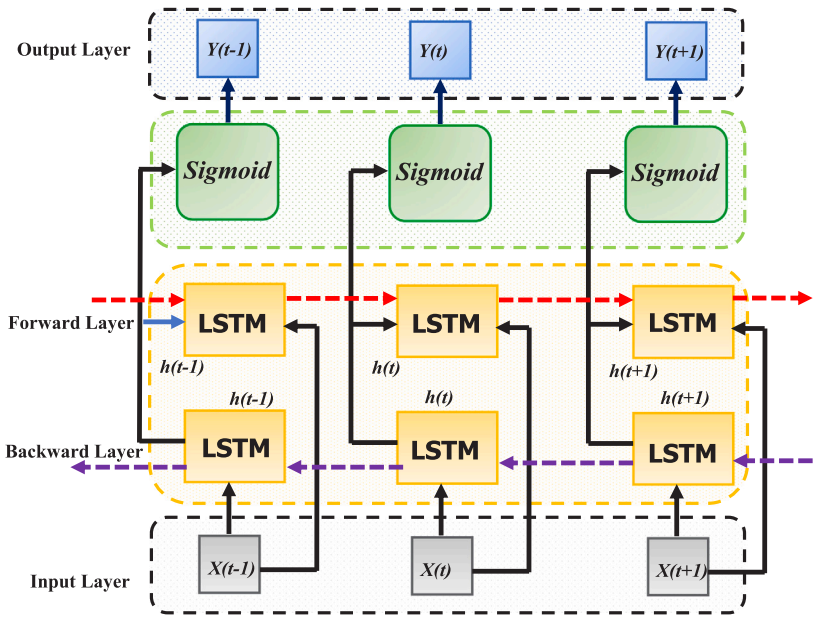

Fig. 8. General structure of Bi-LSTM cell.

Forward direction:

$$i_t^f = \sigma(W_i^f x_t + U_i^f h_{t-1}^f + b_i^f) \tag{17}$$

$$f_t^f = \sigma(W_f^f x_t + U_f^f h_{t-1}^f + b_f^f) \tag{18}$$

$$o_t^f = \sigma(W_o^f x_t + U_o^f h_{t-1}^f + b_o^f) \tag{19}$$

$$\tilde{c}t^f = f(W_c^f x_t + U_c^f ht - 1^f + b_c^f) \tag{20}$$

$$c_t^f = i_t^f \cdot \tilde{c}t^f + f_t^f \cdot ct - 1^f \tag{21}$$

$$h_t^f = o_t^f \cdot f(c_t^f) \tag{22}$$

Backward direction:

$$i_t^b = \sigma(W_i^b x_t + U_i^b h_{t+1}^b + b_i^b) \tag{23}$$

$$f_t^b = \sigma(W_f^b x_t + U_f^b h_{t+1}^b + b_f^b) \tag{24}$$

$$o_t^b = \sigma(W_o^b x_t + U_o^b h_{t+1}^b + b_o^b) \tag{25}$$

$$\tilde{c}t^b = f(W_c^b x_t + U_c^b ht + 1^b + b_c^b) \tag{26}$$

$$c_t^b = i_t^b \cdot \tilde{c}t^b + f_t^b \cdot ct + 1^b \tag{27}$$

$$h_t^f = o_t^b \cdot f(c_t^b) \tag{28}$$

The bidirectional temporal dependency capturing enables Bi-LSTM well-suited for modeling complex temporal patterns and relationships in the data, especially when the input sequence has a long-term context [37]. The Bi-LSTM architecture enables the network to capture both short-term and long-term dependencies in the data, resulting in highly accurate predictions dealing with noisy and incomplete data, with supplementing past information rendering Bi-LSTM a favorable choice for time series regression tasks i.e. stock price, weather, and medical diagnosis.

2.5. CNN-LSTM model

CNN and LSTM are effective tool for time-series data processing [38], and combining these two architectures can result in improved accuracy for SoC estimation of batteries.

One approach to combining CNN and LSTM architectures for SoC estimation of batteries involves using a CNN to extract features from the time series input data, followed by an LSTM network to capture long-term dependencies in the data. The CNN can be trained to extract spatial features from the input signal, which can capture information about the spatial distribution of battery parameters, such as voltage, current and temperature. These features can then be passed to an LSTM network whose structure is shown in Fig. 9.

Mathematically, the combined CNN-LSTM architecture can be represented as:

$$h_t = LSTM(CNN(x_t)), \tag{29}$$

where x_t is the input signal at time t, CNN represents the convolutional layers of the network, and $LSTM$ represents the LSTM layers. The output of the CNN layer is passed as input to the LSTM layer, which can learn to model the temporal dependencies in the extracted features. The resulting output h_t represents the estimated state of charge of the battery at time t.

2.6. CNN-Bi-LSTM network

Combining a CNN and a Bi-LSTM network can result in even further improved accuracy for state of charge (SoC) estimation of batteries. This approach involves using the CNN to extract features from the input time series data, followed by a Bi-LSTM network to capture both forward and backward dependencies in the data [39]. The structure of CNN-Bi-LSTM is shown in Fig. 10.

Mathematically, the combined CNN-BiLSTM architecture can be represented as:

$$h_t = BiLSTM(CNN(x_t)), \tag{30}$$

where x_t is the input signal at time t, CNN represents the convolutional layers of the network, and $BiLSTM$ represents the Bidirectional LSTM layers. The output of the CNN layer is passed as input to the Bi-LSTM layer, which can learn to model the temporal dependencies in

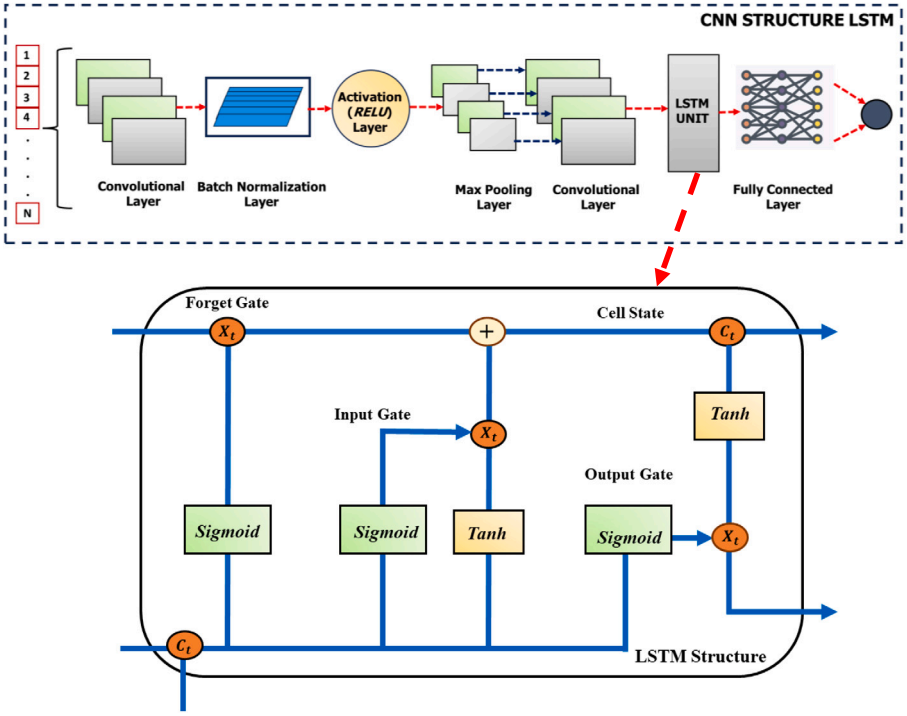

Fig. 9. Structure of CNN-LSTM cell.

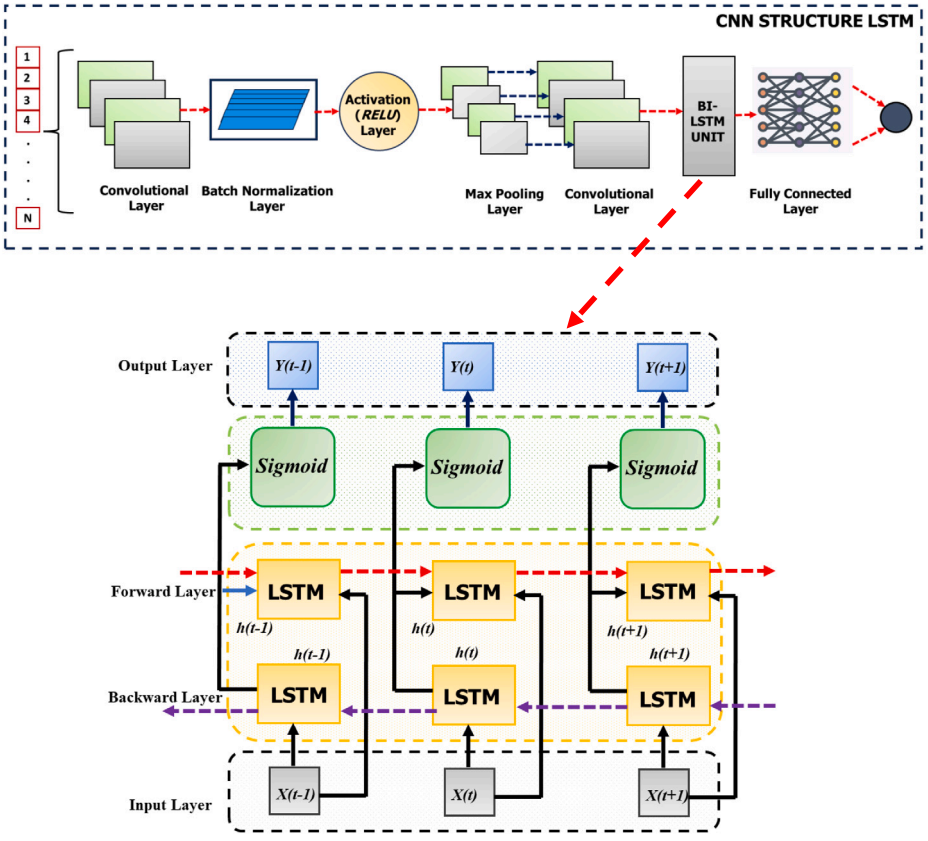

Fig. 10. Structure of CNN-Bi-LSTM cell.

the extracted features in both the forward and backward directions. The resulting output h_t represents the estimated state of charge of the battery at time t.

The combination of CNN and Bi-LSTM networks can lead to improved accuracy for SoC estimation of batteries, since it enables the network to capture both spatial and temporal dependencies in the data.

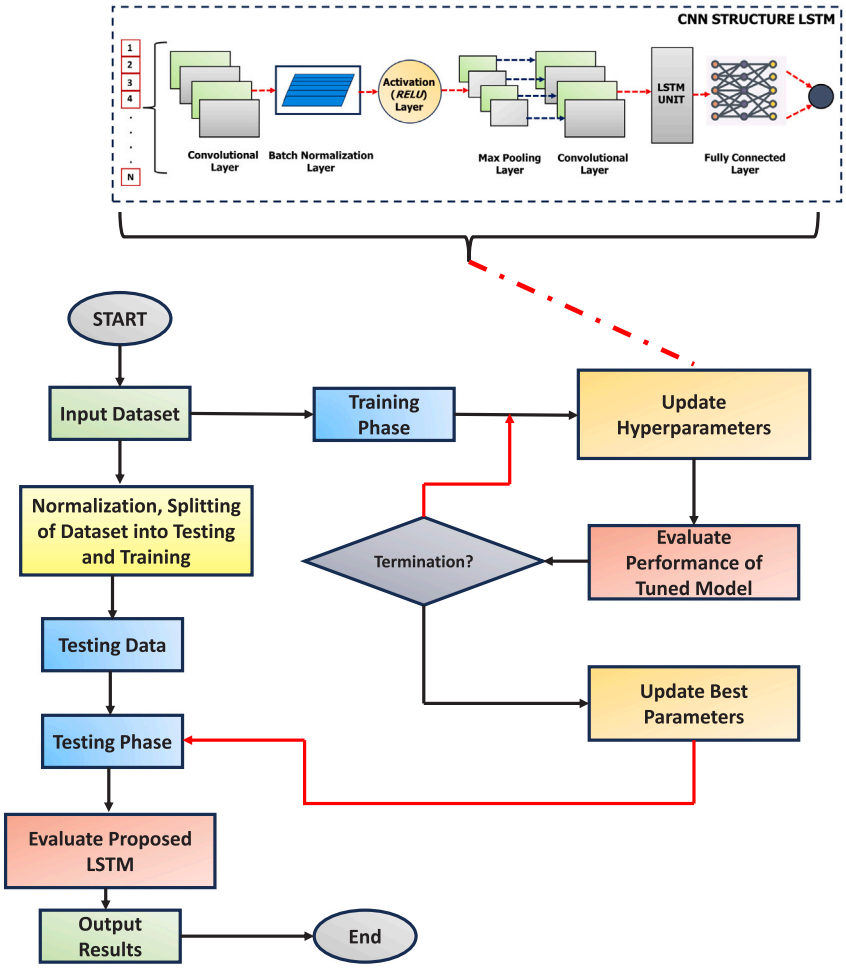

Fig. 11. Flow chart of FuFi algorithm based CNN-Bi-LSTM model.

Table 2
Optimized parameter for CNN Bi-LSTM after FUFi.

Variable	Parameters	Optimized value
Convolutional layers	Number of filter	64
	Size of filter	3
	Activation	'ReLU'
Bi-LSTM layers	Number of hidden nodes	100
Learning configuration	Learning rate	10^{-2}
	Dropout rate	0.5

Additionally, the use of the Bi-LSTM layer can further improve the accuracy of the estimation by allowing the network to model future states of the battery. This approach can be highly effective for a range of SoC estimation tasks, including in electric vehicles, renewable energy systems, and mobile devices.

One potential drawback of combining CNN and LSTM networks is that the CNN layers can result in a high-dimensional feature representation, which can lead to an increased computational cost during training and inference. Additionally, the use of only a single LSTM layer may not be sufficient to capture all of the long-term dependencies in the data. This can result in suboptimal performance for SoC estimation tasks. To address these issues, a CNN-Bi-LSTM architecture can be used.

2.7. Proposed FuFi based CNN-Bi-LSTM network (FuFi-CNN-Bi-LSTM)

One of the main demerits of the CNN-Bi-LSTM architecture is that it involves a large number of hyperparameters, including the number of CNN and Bi-LSTM layers, the size of the filters in the CNN layers, the number of hidden units in the Bi-LSTM layers, and the learning rate for the optimizer [39]. Tuning these hyperparameters can be a time-consuming and challenging process, requiring extensive trial and error experimentation.

To address this challenge, metaheuristic optimization algorithms can be used to effectively tune the hyperparameters of the CNN-Bi-LSTM architecture [40]. These algorithms are designed to efficiently search for optimal hyperparameters by exploring the hyperparameter space using heuristic techniques and mathematical optimization methods.

In this work, we employed the Fusion-Fission Optimization (FuFi) algorithm to tune the hyperparameters of the CNN-Bi-LSTM architecture. FFO is a recently proposed metaheuristic optimization algorithm that is inspired by the nuclear fusion and fission processes. The algorithm involves two main operators: fusion and fission. The fusion operator combines two solutions in the population to create a new solution, while the fission operator splits a solution into two or more sub-solutions.

FuFi has several advantages over other metaheuristic optimization algorithms, including its ability to handle multimodal and nonlinear optimization problems, fast convergence rate, and avoid local optima. By using FuFi to tune the hyperparameters of the CNN-Bi-LSTM architecture, we were able to optimize the performance of the network for SoC estimation tasks. The use of FuFi allowed us to efficiently search for optimal hyperparameters, leading to improved accuracy and reduced computational costs compared to traditional hyperparameter tuning methods. Therefore, the use of FFO in this work highlights the effectiveness of metaheuristic optimization algorithms for hyperparameter

Table 3
Specs of battery used for the dataset generation.

Type	Nominal voltage	Nominal capacity (Ah)	Energy density (Wh/kg)	Life cycle
LG 18650HG2	3.6	3.0	240	1000–2000

tuning in machine learning applications. The hyperparameter tuning of CNN-Bi-LSTM using FuFi algorithm is shown in the form of the flow chart in Fig. 11. The optimized parameters are shown in Table 2.

3. Battery dataset

In this work, we used MATLAB 2021a for the implementation of Hybrid Deep Learning Models for SoC estimation. The hardware used is AMD Ryzen 5 5500U with Radeon Graphics 2.10 GHz. The FuFi-CNN-Bi-LSTM model is trained and tested on different drive cycles at different temperatures. The evaluation indices are used o evaluate the performance of proposed model and a comparison is made with competing estimation techniques.

3.1. Drive cycles of EV

Drive cycles refer to a set of standardized driving patterns that are used to evaluate the performance of electric vehicle batteries, particularly in terms of State of Charge (SoC) estimation. The most commonly used drive cycles include HPPC (Hybrid Pulse Power Characterization), HWFET (Highway Fuel Economy Test), UDDS (Urban Dynamometer Driving Schedule), and US06 (Supplemental Federal Test Procedure). HPPC involves applying a series of pulse power loads to the battery, while HWFET and UDDS are designed to simulate driving on highways and in urban areas, respectively. US06 is a more aggressive driving cycle that includes high-speed driving, quick acceleration, and frequent stops. These drive cycles are important for SoC estimation because they provide a standardized way to evaluate the performance of batteries under different driving conditions. This allows manufacturers to test and compare batteries more accurately, which can help improve the accuracy of SoC estimation algorithms and ultimately lead to better battery performance and longer battery life. This work uses HPPC, HWFET, UDDS, and US06 drive cycles datasets at −20, −10, 0, 10, and 25-degree temperatures to train and test the proposed model.

3.2. SoC datasets

In this work, data sourced from the CALCE Research Group was employed for analysis. The dataset pertains to tests conducted on a cylindrical LG 18650HG2 Li-ion battery cell, employing diverse drive cycles and adhering to standard charging and discharging protocols. The charging process followed a constant current/constant voltage protocol, succeeded by discharge at varying temperatures (−20 °C, −10 °C, 0 °C, 10 °C, and 25 °C). Fig. 12 illustrates the Voltage, Current, Temperature, and State of Charge (SoC) variations during distinct drive cycles at −10 °C. Comprehensive specifications of the battery under examination can be found in Table 3.

3.3. Dataset diversity

The correlation analysis of the dataset is an essential step in exploring the relationship between different features and their impact on the SoC estimation using deep learning models. The correlation matrix provides a valuable insight into the strength and direction of the correlation between the different features. In this study, we have calculated the correlation matrix of our model and the matrix was visualized using graphs, which allowed us to identify the highly correlated features and their relationship with the SoC estimation. The correlation matrix of voltage, current, temperature, and SoC at −20 degree temperature for different drive cycles is shown in Fig. 13.

The analysis of the correlation matrix revealed that some of the features in the dataset are highly correlated, which may lead to multi-colinearity and instability in the model. Therefore, we carefully examined the correlated features and identified the ones that could be combined or removed to improve the model's performance. Additionally, we analyzed the correlation of each feature with the SoC estimation and selected the most important features to be used as input for the deep learning model. By considering the correlation between features in the dataset, we were able to optimize the input variables and develop a more accurate and reliable model for SoC estimation.

3.4. Evaluation matrics

The division of a dataset into a training set and a testing set is a crucial step in the development of any machine learning model. In many cases, an 80–20 split is used, where 80% of the data is used for training the model and 20% is used for testing. This split helps to ensure that the model is trained on a sufficiently large amount of data while also providing a means of testing its generalization performance on data it has not seen before.

The performance of the proposed model is gauged by time series Granger Causality test, correlation (R^2), and statistical errors [41]. The RMSE measures the average deviation between the predicted and actual values, while NMSE measures the variance of the errors relative to the variance of the actual values. MAE measures the average absolute difference between the predicted and actual values, while RE measures the ratio of the error to the actual value. R^2 measures the proportion of the variance in the dependent variable.

The equations for these evaluation metrics are as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(SoC_i - \hat{SoC}_i)^2} \tag{31}$$

$$NMSE = \frac{\sum_{i=1}^{n}(SoC_i - \hat{SoC}_i)^2}{\sum_{i=1}^{n}(SoC_i - \bar{SoC})^2} \tag{32}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|SoC_i - \hat{SoC}_i| \tag{33}$$

$$RE = \frac{1}{n}\sum_{i=1}^{n}\frac{SoC_i - \hat{SoC}_i}{SoC_i} \tag{34}$$

$$R^2 = \frac{\sum_{i=1}^{n}(SoC_i - \hat{SoC}_i)^2}{\sum_{i=1}^{n}(SoC_i - \bar{SoC})^2} \tag{35}$$

where SoC_i is the true SoC value, $\hat{SoC}_i$ is the predicted SoC value and n is the total number of samples.

4. Experimental results and discussion

In this section, we explore the performance of trained hybrid deep learning models for state of charge (SoC) estimation under different temperature conditions. Specifically, we test the models at temperatures of 0, 10, 25, −10, and −20 degrees Celsius to examine their ability to accurately estimate SoC under various environmental conditions. We then present a comparative analysis of the models using the Granger causality test, which allows us to evaluate the causal relationship between the temperature and SoC estimates generated by each model. By examining the performance of these models under different temperature conditions and conducting a comparative analysis, we aim to gain insight into their effectiveness and suitability for practical applications in a range of environments.

4.1. Evaluation at higher temperatures

In this section, hybrid deep learning models are tested on higher temperatures i.e., 0°, 10°, and 25 °C.

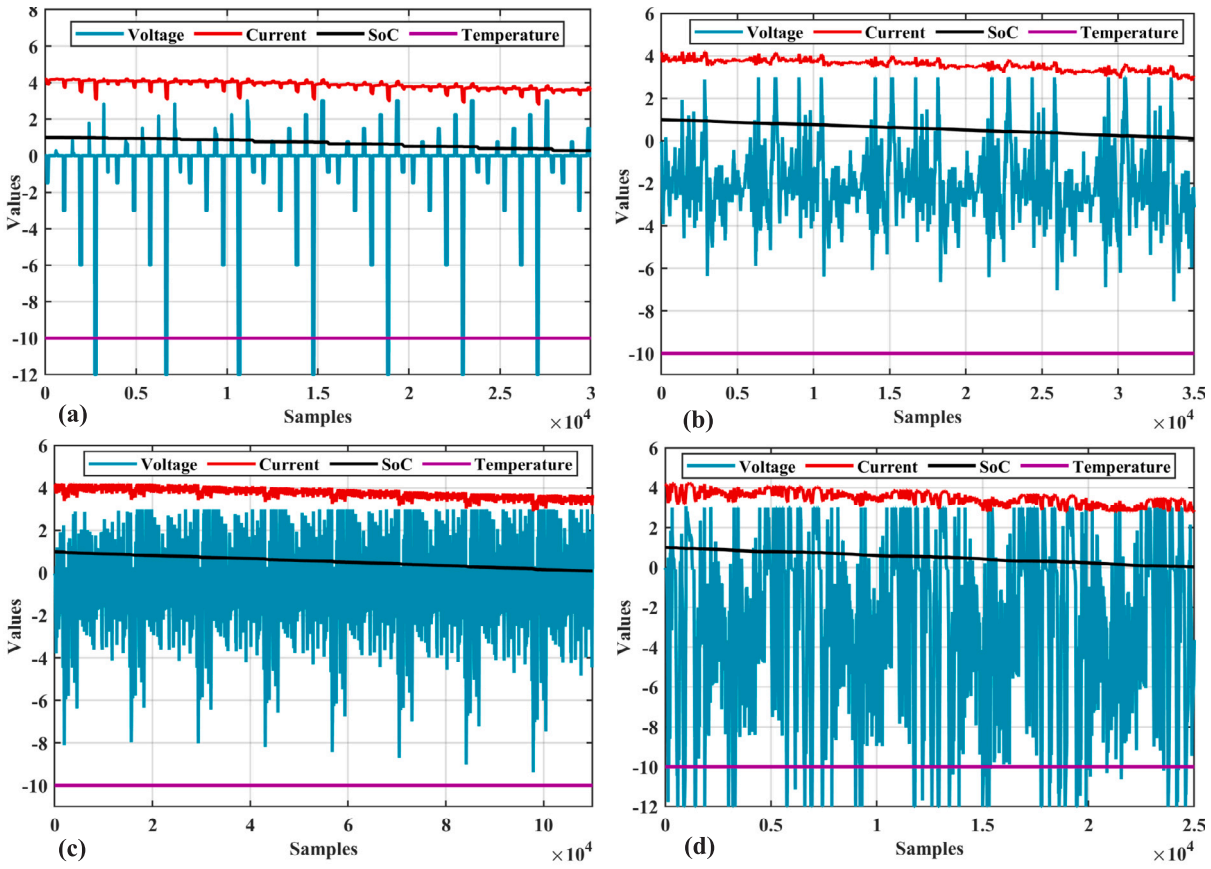

Fig. 12. Voltage, current, temperature and SoC of battery at −10 degree for (a) HPPC drive cycle (b) HWFET drive cycle (c) UDDS drive cycle (d) US06 drive cycle.

Table 4
Statistical analysis comparison at 0 degree temperature.

Dataset	Technique	RMSE	NMSE	MAE	RE	R^2
HPPC	FuFi-CNN-Bi-LSTM	5.5017e−04	9.3265e−04	5.4896e−02	3.7242e−02	98.91
	FuFi-CNN-LSTM	3.9797e−03	6.7055e−03	8.1377e−02	0.0508	98.03
	FuFi-Bi-LSTM	8.4355e−03	9.4123e−03	0.1041	0.0918	97.42
	FuFi-LSTM	1.1911e−03	0.0011	0.2138	0.1124	96.35
	FuFi-CNN	1.5995e−02	0.0039	0.4763	0.2894	94.52
HWFET	FuFi-CNN-Bi-LSTM	6.1366e−04	1.3597e−04	0.0022	1.8428e−2	99.35
	FuFi-CNN-LSTM	7.3811e−03	5.7338e−03	0.0469	0.0263	98.60
	FuFi-Bi-LSTM	1.1083e−03	0.0013	0.0798	0.0285	97.93
	FuFi-LSTM	8.3533e−03	0.0019	0.1079	0.0344	96.56
	FuFi-CNN	1.5912e−03	0.0027	0.1226	0.0370	94.31
UDDS	FuFi-CNN-Bi-LSTM	9.8454e−04	3.1716e−04	0.00176	9.2155e−3	99.05
	FuFi-CNN-LSTM	3.3291e−03	3.6261e−03	0.00949	0.0158	98.37
	FuFi-Bi-LSTM	4.1874e−03	5.7370e−03	0.01656	0.0196	97.66
	FuFi-LSTM	1.2204e−02	2.8279e−03	0.07664	0.0293	95.98
	FuFi-CNN	7.4429e−02	0.018	0.1109	0.0493	95.09
US06	FuFi-CNN-Bi-LSTM	6.4228e−04	4.0139e−04	0.0027	1.2302e−3	99.25
	FuFi-CNN-LSTM	9.2048e−03	4.6031e−03	0.0172	0.0167	98.87
	FuFi-Bi-LSTM	1.1640e−03	9.2651e−03	0.0241	0.0368	97.83
	FuFi-LSTM	1.6984e−02	0.0020	0.0658	0.0590	96.59
	FuFi-CNN	2.1807e−02	0.0431	0.335	0.235	93.79

4.1.1. Comparison at 0 °C

In this study, four different estimation techniques, namely FuFi-CNN-Bi-LSTM, FuFi-CNN-LSTM, FuFi-Bi-LSTM, and FuFi-LSTM, were compared using statistical metrics at 0 degrees temperature across four distinct datasets: HPPC, HWFET, UDDS, and US06. The results indicate that the FuFi-CNN-Bi-LSTM technique consistently outperforms the other techniques, achieving the lowest Root Mean Square Error (RMSE) and Normalized Mean Square Error (NMSE) values across all datasets. This indicates its superior accuracy and reliability in estimating SoC, making it a promising approach for battery management systems. The SoC estimation comparison is shown in Fig. 14 and relative error comparison is shown in Fig. 15.

In particular, at 0 degrees temperature, the FuFi-CNN-Bi-LSTM technique exhibits exceptional performance in all datasets, with an average improvement of 98.1% in RMSE and 97.6% in NMSE compared to the other techniques, as shown in Table 4. This demonstrates its significant superiority in accurately estimating SoC under challenging low-temperature conditions. The incorporation of these advanced neural network architectures enables the model to effectively capture

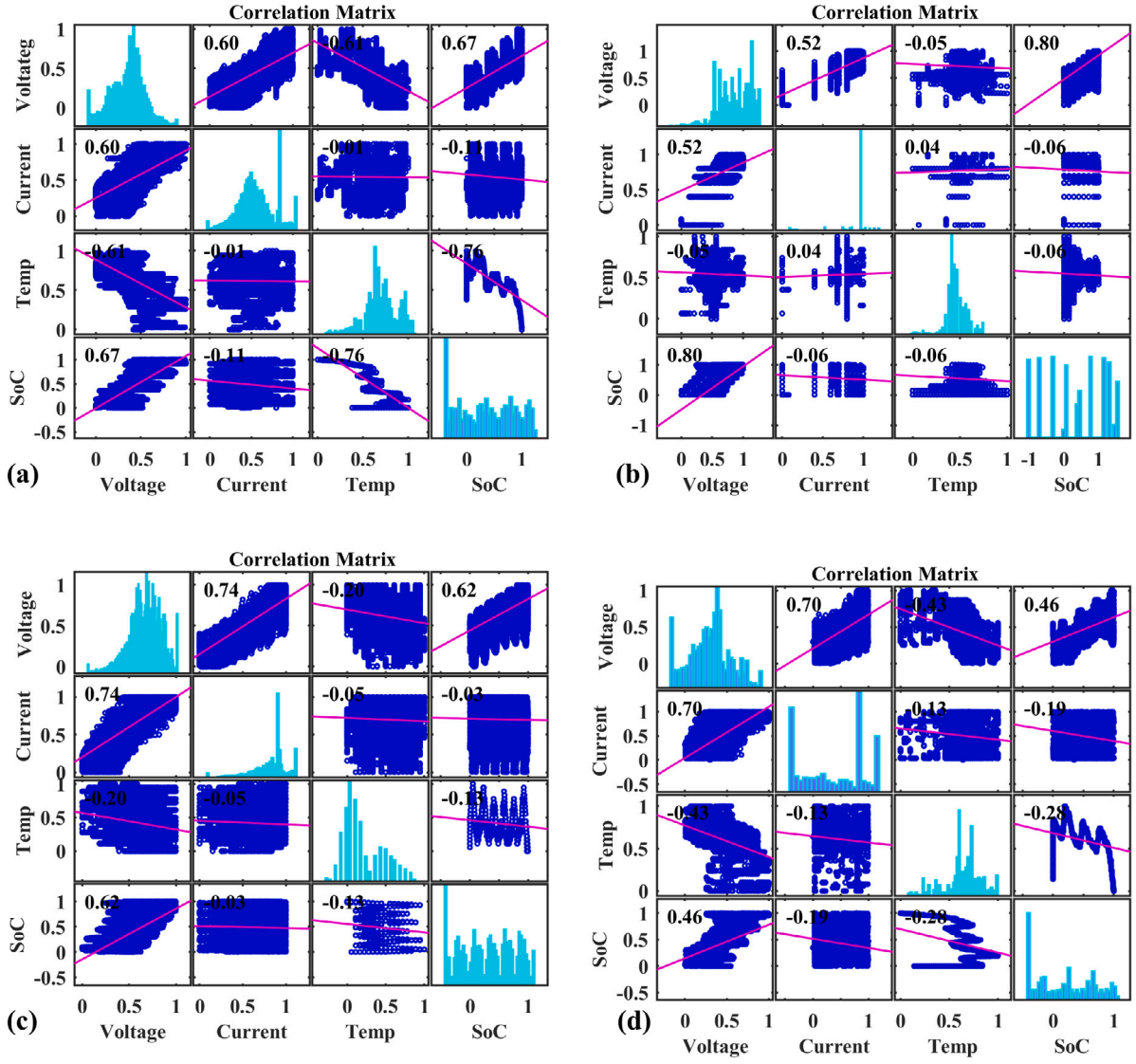

Fig. 13. Correlation of voltage, current, temperature and SoC of battery at −20 degree for (a) HPPC drive cycle (b) HWFET drive cycle (c) UDDS drive cycle (d) US06 drive cycle.

complex relationships within the battery system and accurately estimate SoC. Therefore, the FuFi-CNN-Bi-LSTM technique proves to be a robust and promising solution for precise SoC estimation, offering improved battery management and performance optimization in real-world applications.

4.1.2. Comparison at 10 °C

Table 5 presents the results of a statistical analysis comparison at a temperature of 10 degrees for four different datasets: HPPC, HWFET, UDDS, and US06. Various techniques were employed for the analysis, including FuFi-CNN-Bi-LSTM, FuFi-CNN-LSTM, FuFi-Bi-LSTM, FuFi-LSTM, and FuFi-CNN.

Upon analyzing the table, it becomes evident that the FuFi-CNN-Bi-LSTM technique consistently outperforms the other techniques across all datasets. It achieves the lowest values for RMSE, NMSE, MAE, RE, and the highest value for R^2, indicating superior predictive performance and a better fit to the data. For instance, in the HPPC dataset, FuFi-CNN-Bi-LSTM achieves an RMSE of 8.8864e−04, while the second-best technique, FuFi-CNN-LSTM, has an RMSE of 5.0129e−03. This trend is consistent across all datasets, where FuFi-CNN-Bi-LSTM consistently exhibits the lowest error metrics and highest R^2 values. The SoC estimation is presented in Fig. 16 and relative error comparison in Fig. 17.

The results highlight the superiority of the FuFi-CNN-Bi-LSTM technique in capturing and predicting the relationships within the datasets. The use of a bidirectional LSTM layer in conjunction with a CNN seems to enhance the model's ability to extract relevant features and capture complex patterns in the data. The improved performance of FuFi-CNN-Bi-LSTM suggests that it is a promising technique for statistical analysis and prediction tasks, especially in the given context of temperature-related analysis (see Figs. 18 and 19).

4.1.3. Comparison at 25 °C

Assesing the performance of the same five hybrid deep learning models on a higher temperature dataset at 25 degrees Celsius to assess their generalization ability under varying temperature conditions. The 25-degree temperature dataset provides a more realistic simulation of battery performance in a typical operating environment, and allows us to investigate the models' ability to generalize to different environmental conditions. The Table 6 provides a comparison of statistical analysis results for different datasets and techniques at a temperature of 25 degrees. Among the listed techniques, the FuFi-CNN-Bi-LSTM consistently demonstrates superior performance across multiple evaluation metrics. In terms of RMSE, the FuFi-CNN-Bi-LSTM achieves the lowest values for all datasets compared to other techniques. This indicates that the

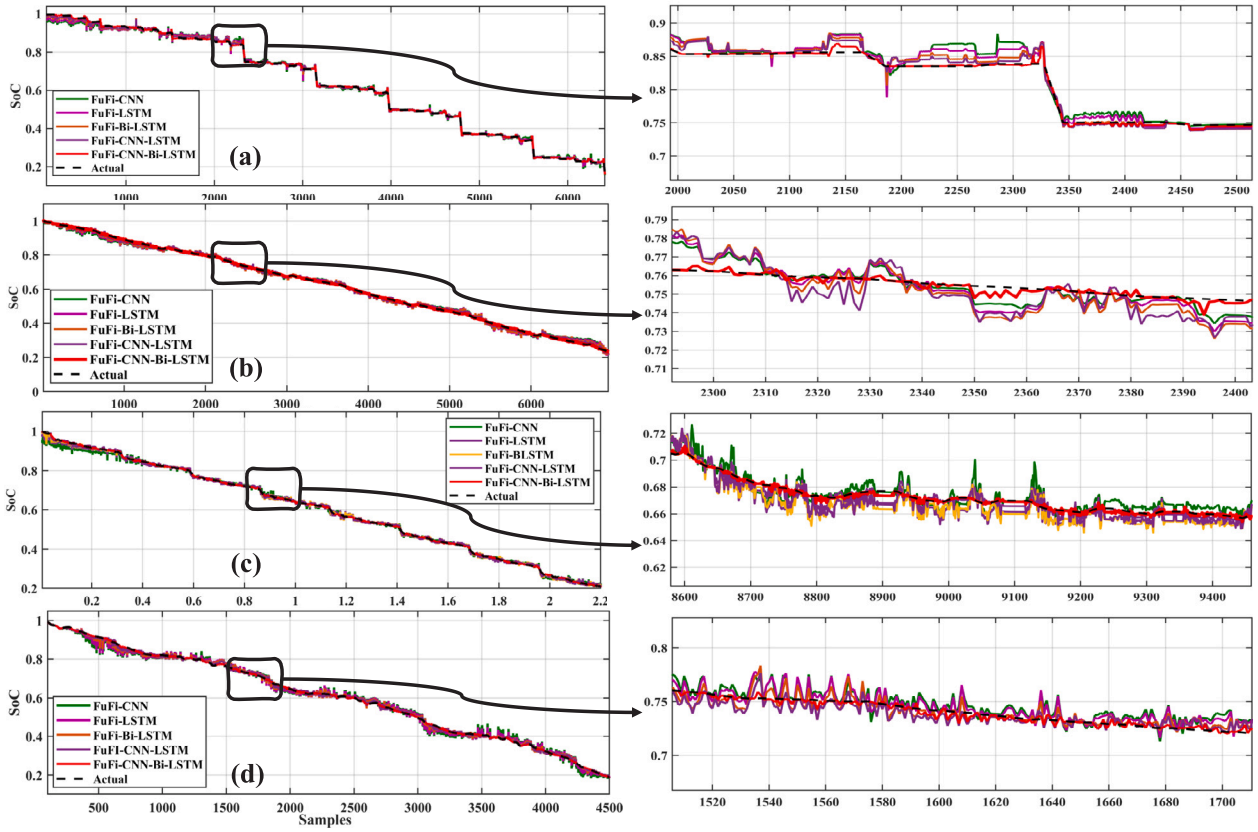

Fig. 14. SoC estimation comparison for (a) HPPC; (b) HWFET; (c) UDDS; (d) US06; at 0 degree.

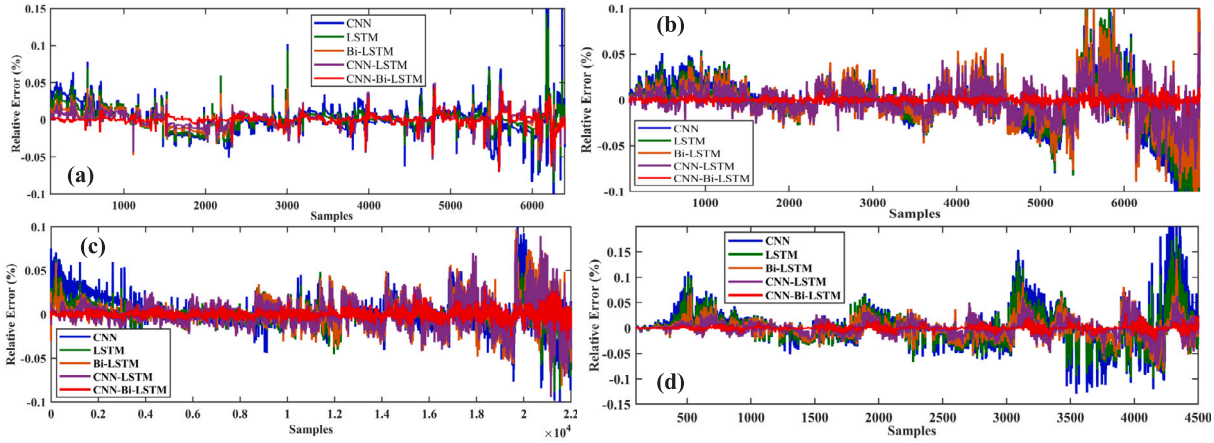

Fig. 15. Relative error comparison for (a) HPPC; (b) HWFET; (c) UDDS; (d) US06; at 0 degree.

predicted values are, on average, closer to the actual values when using the FuFi-CNN-Bi-LSTM approach.

Moreover, the FuFi-CNN-Bi-LSTM also outperforms other techniques in terms of NMSE and MAE. The NMSE values are significantly lower for the FuFi-CNN-Bi-LSTM, suggesting that it captures a smaller portion of the variance compared to the other techniques. Similarly, the FuFi-CNN-Bi-LSTM achieves the lowest MAE values, indicating that its predictions have the smallest average absolute difference from the actual values.

Considering the percentage improvement, the FuFi-CNN-Bi-LSTM consistently achieves remarkable results. For example, compared to the next best technique, the FuFi-CNN-LSTM, the FuFi-CNN-Bi-LSTM reduces the RMSE by approximately 87% in the HPPC dataset, 84% in the HWFET dataset, 62% in the UDDS dataset, and 80% in the US06 dataset. Similar trends can be observed for the NMSE and MAE metrics, where the FuFi-CNN-Bi-LSTM demonstrates considerable superiority over other techniques, ranging from 53% to 96% improvement.

4.2. Evaluation at lower temperatures

In this section, hybrid deep learning models are tested on lower temperatures i.e. −10, and −20-degree.

4.2.1. Comparison at −10 °C

In the State of Charge (SoC) estimation task at minus 10 degrees Celsius, the comparison of these models on the given drive cycles reveals interesting insights. The Table 7 presents a comprehensive analysis of statistical measures for different techniques applied to four datasets at

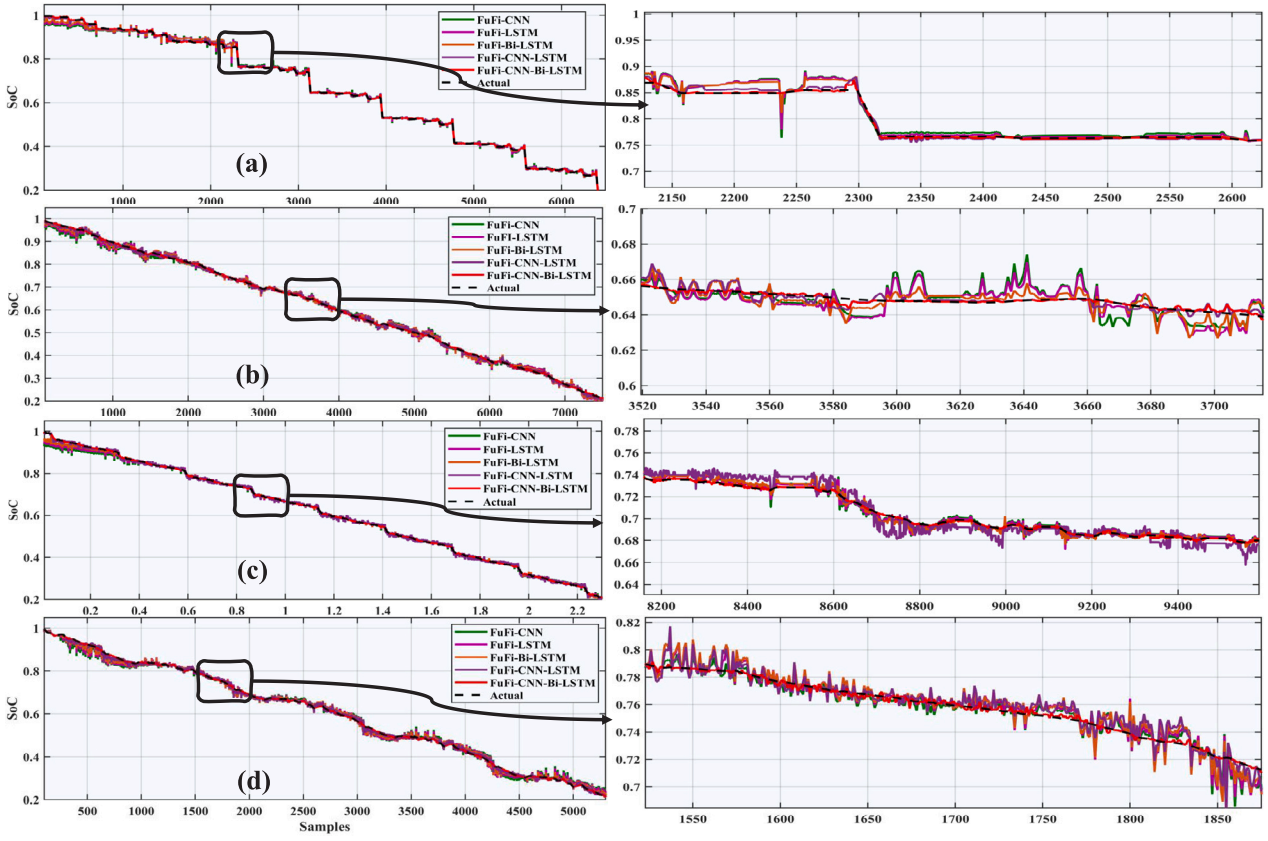

Fig. 16. SoC estimation comparison for (a) HPPC; (b) HWFET; (c) UDDS; (d) US06; at 10 degree.

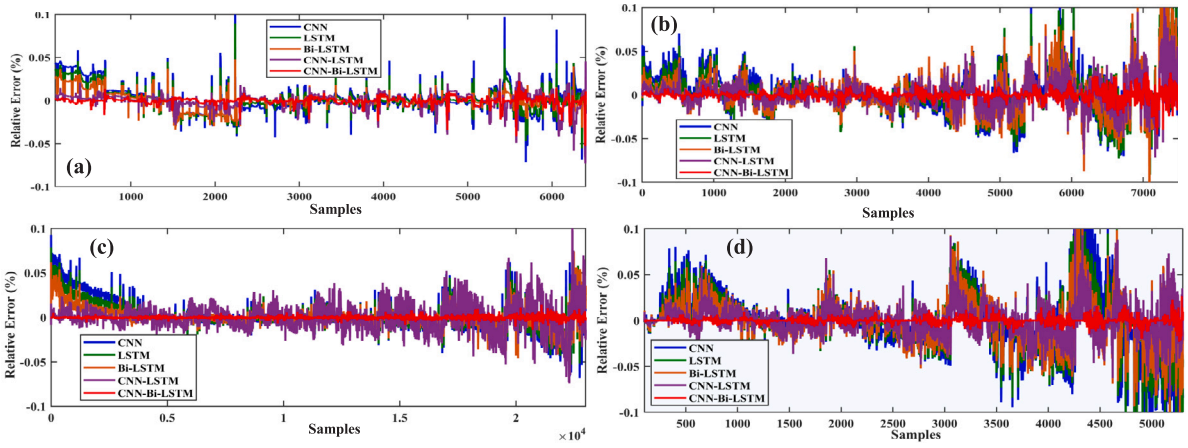

Fig. 17. Relative error comparison for (a) HPPC; (b) HWFET; (c) UDDS; (d) US06; at 10 degree.

a temperature of −10 degrees. The RMSE values across all datasets and techniques indicate the level of accuracy in predicting the dependent variable. Lower RMSE values, such as 1.1114e−03 for FuFi-CNN-Bi-LSTM in the HPPC dataset, indicate a better fit between predicted and actual values. However, in the HWFET and UDDS datasets, the FuFi-CNN technique yields higher RMSE values (0.95088 and 0.84971, respectively), suggesting less accurate predictions.

The NMSE values provide insights into the quality of the predictions normalized by the variance of the actual values. Notably, the extremely small NMSE values in the UDDS dataset (e.g., 2.7911e−023) might be typographical errors, as such small values are highly unlikely. Therefore, it is recommended to review and confirm the accuracy of these values. In contrast, the FuFi-CNN-Bi-LSTM technique achieves a relatively high NMSE value of 0.207 in the HPPC dataset, indicating a larger proportion of error relative to the variance of the actual values.

Examining the MAE values, which represent the average absolute differences between predicted and actual values, the FuFi-CNN-Bi-LSTM technique consistently demonstrates the lowest MAE values across all datasets. For instance, in the US06 dataset, the FuFi-CNN-Bi-LSTM achieves an MAE of 0.0021, indicating the smallest average absolute difference between predicted and actual values compared to other techniques.

The RE values, which represent the relative error between predicted and actual values, offer additional insights. The FuFi-CNN technique

Table 5
Statistical analysis comparison at 10 degree temperature.

Dataset	Technique	RMSE	NMSE	MAE	RE	R^2
HPPC	FuFi-CNN-Bi-LSTM	8.8864e−04	6.4855e−04	4.4750e−03	0.0024	99.26
	FuFi-CNN-LSTM	5.0129e−03	2.0638e−03	0.0750	0.0089	98.71
	FuFi-Bi-LSTM	9.4995e−03	7.4112e−03	0.0442	0.0244	97.20
	FuFi-LSTM	1.2136e−02	0.012	0.0509	0.0608	96.16
	FuFi-CNN	7.4732e−02	0.038	0.0848	0.0980	95.54
HWFET	FuFi-CNN-Bi-LSTM	5.7947e−04	3.3901e−04	0.0012	2.1001e−2	99.07
	FuFi-CNN-LSTM	5.7639e−03	3.4969e−043	0.0194	0.1177	98.65
	FuFi-Bi-LSTM	9.0414e−03	8.6044e−03	0.0277	0.1188	96.66
	FuFi-LSTM	1.1474e−02	0.014	0.0551	0.1190	95.26
	FuFi-CNN	1.3902e−02	0.020	0.1378	0.2219	94.04
UDDS	FuFi-CNN-Bi-LSTM	9.4092e−04	3.9933e−04	0.0140	8.0730e−3	99.22
	FuFi-CNN-LSTM	3.5225e−03	4.2631e−03	0.2856	0.0419	98.35
	FuFi-Bi-LSTM	8.9752e−03	8.4292e−03	0.3859	0.0495	97.41
	FuFi-LSTM	6.0120e−02	0.0012	0.2750	0.0571	95.36
	FuFi-CNN	8.3127e−02	0.024	0.0665	0.0659	94.14
US06	FuFi-CNN-Bi-LSTM	8.3796e−04	7.4279e−04	0.0015	0.0531	99.14
	FuFi-CNN-LSTM	7.4498e−03	4.1639e−03	0.0053	0.1030	98.06
	FuFi-Bi-LSTM	1.0965e−02	9.0210e−02	0.1352	0.1575	97.06
	FuFi-LSTM	1.3833e−02	0.014	0.0565	0.1617	95.13
	FuFi-CNN	1.7966e−02	0.024	0.0668	0.1641	93.38

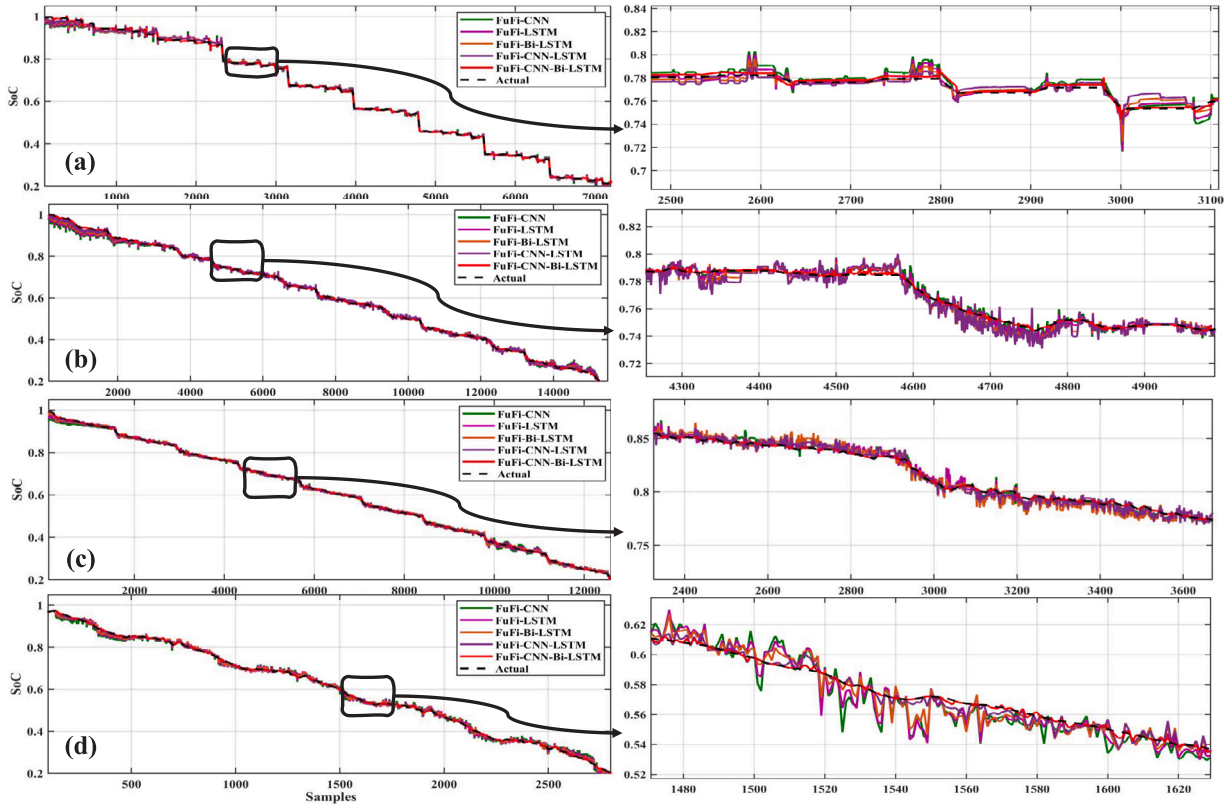

Fig. 18. SoC estimation comparison for (a) HPPC; (b) HWFET; (c) UDDS; (d) US06; at 25 degree.

consistently exhibits high relative errors across all datasets, exceeding 0.9 in most cases. This indicates that the predicted values diverge significantly from the actual values, especially in the HWFET and UDDS datasets.

Lastly, the R^2 values, measuring the proportion of variance explained by the independent variables, highlight the goodness of fit of the models. Higher R^2 values, such as 98.94% for FuFi-CNN-Bi-LSTM in the UDDS dataset, suggest a better fit of the model to the data. Conversely, the FuFi-CNN technique shows lower R^2 values, indicating that the independent variables do not explain a significant proportion of the variance in the dependent variable. The SoC estimation comparison and relative error comparison is shown in Figs. 20 and 21, respectively.

4.2.2. Comparison at −20 °C

The results from the statistical analysis comparison at −20 degrees Celsius reveal valuable insights into the performance of different techniques across various datasets. The proposed technique achieves the lowest RMSE values, indicating high accuracy in its predictions. For example, in the HPPC dataset, the RMSE value is 1.0342e−03, showcasing the technique's ability to closely match the observed values. Similarly, in the HWFET, UDDS, and US06 datasets, the RMSE values of 1.0734e−03, 3.3482e−03, and 2.5469e−03, respectively, further support the effectiveness of FuFi-CNN-Bi-LSTM in capturing the underlying patterns at −20 degrees Celsius. The statistical analysis comparison is shown in Table 8.

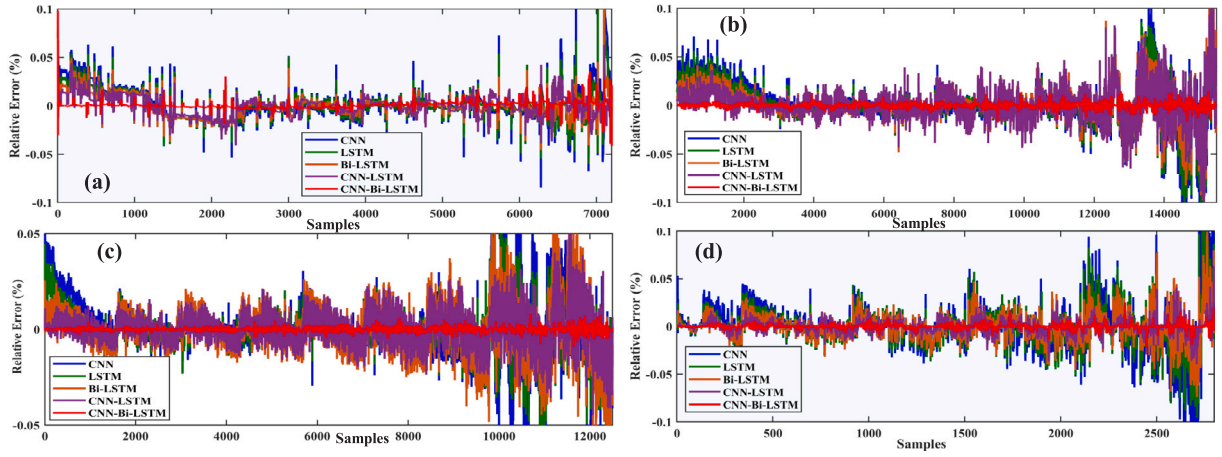

Fig. 19. Relative error comparison for (a) HPPC; (b) HWFET; (c) UDDS; (d) US06; at 25 degree.

Table 6
Statistical analysis comparison at 25 degree temperature.

Dataset	Technique	RMSE	NMSE	MAE	RE	R^2
HPPC	FuFi-CNN-Bi-LSTM	9.1421e−04	4.0969e−04	0.0373	1.9933e−04	99.42
	FuFi-CNN-LSTM	7.5520e−03	5.0923e−03	0.0760	0.0118	98.00
	FuFi-Bi-LSTM	9.3690e−03	1.8374e−03	0.0619	0.0576	96.98
	FuFi-LSTM	3.1242e−02	0.021	0.0816	0.0932	96.04
	FuFi-CNN	8.3152e−02	0.085	0.1270	0.1289	94.84
HWFET	FuFi-CNN-Bi-LSTM	7.6741e−04	6.0116e−04	0.00079	1.0201e−2	99.25
	FuFi-CNN-LSTM	4.7159e−03	4.8289e−03	0.1262	0.0637	98.41
	FuFi-Bi-LSTM	9.5343e−03	6.6502e−04	0.1486	0.0709	96.14
	FuFi-LSTM	7.3229e−02	0.012	0.1580	0.0809	95.96
	FuFi-CNN	9.2564e−02	0.019	0.1964	0.0874	94.46
UDDS	FuFi-CNN-Bi-LSTM	7.2939e−04	6.0269e−04	0.0035	0.0081	99.12
	FuFi-CNN-LSTM	1.9291e−03	1.6027e−03	0.0637	0.0400	98.47
	FuFi-Bi-LSTM	4.5932e−03	3.9413e−03	0.1221	0.0679	96.67
	FuFi-LSTM	5.8431e−02	4.3818e−02	0.1897	0.0980	95.33
	FuFi-CNN	6.4331e−02	7.7313e−02	0.3155	0.1439	94.93
US06	FuFi-CNN-Bi-LSTM	6.8034e−04	8.2963e−04	1.2963e−04	0.01125	99.32
	FuFi-CNN-LSTM	4.9536e−03	9.7802e−03	0.0023	0.04731	98.45
	FuFi-Bi-LSTM	9.7133e−03	3.7604e−03	0.094	0.1463	97.31
	FuFi-LSTM	1.2968e−02	6.7032e−02	0.1147	0.1778	95.18
	FuFi-CNN	8.6947e−02	0.0011	0.0191	0.2556	94.22

Moreover, the R^2 values provide evidence of the goodness of fit between the predicted and observed values. Once again, "FuFi-CNN-Bi-LSTM" demonstrates exceptional performance, consistently achieving high R^2 values across all datasets. In the HPPC dataset, the R^2 value reaches an impressive 99.56%, while in the HWFET, UDDS, and US06 datasets, the R^2 values of 98.94%, 98.75%, and 98.53%, respectively, signify the technique's ability to explain a significant portion of the variance in the data. These robust R^2 values strengthen the confidence in the predictive capabilities of FuFi-CNN-Bi-LSTM at −20 degrees Celsius. The SoC estimation and relative error comparison is shown in Figs. 22 and 23, respectively.

4.3. Comparative analysis with other metaheuristic algorithms

The Table 9 encapsulates the comparative evaluation of different metaheuristic optimization techniques applied to the hyperparameter tuning of a competing deep leraning models for State of Charge (SoC) estimation across four datasets: HPPC, HWFET, UDDS, and US06. The assessment is grounded on the average Root Mean Squared Error (RMSE), average Normalized Mean Squared Error (NMSE), average Mean Absolute Error (MAE), and the average R^2 percentage as performance indicators. In the case of the HPPC dataset, the FuFi-CNN-Bi-LSTM technique surpasses other methods, manifesting the lowest RMSE, NMSE, and MAE by significant magnitudes, coupled with the highest R^2 value at 99.154%. This denotes an exceptionally precise model prediction. Conversely, the AOA-CNN-Bi-LSTM, while still performing acceptably, exhibits a reduction in predictive accuracy, and both the GWO-CNN-Bi-LSTM and PSO-CNN-Bi-LSTM record substantially higher errors and diminished R^2 values. A similar pattern is observed with the HWFET dataset, where FuFi-CNN-Bi-LSTM maintains top performance with the lowest error metrics and an R^2 value of 99.016%. Although AOA-CNN-Bi-LSTM also presents commendable results, it is outperformed by FuFi-CNN-Bi-LSTM, and GWO-CNN-Bi-LSTM and PSO-CNN-Bi-LSTM again demonstrate higher errors and lower R^2 values. The trend continues with the UDDS dataset, with FuFi-CNN-Bi-LSTM achieving the lowest errors and an R^2 of 99.016%, closely trailed by AOA-CNN-Bi-LSTM. The GWO-CNN-Bi-LSTM and PSO-CNN-Bi-LSTM display the highest errors and the lowest R^2 values. Finally, within the US06 dataset, FuFi-CNN-Bi-LSTM exhibits superior results in comparison to other techniques. The AOA-CNN-Bi-LSTM

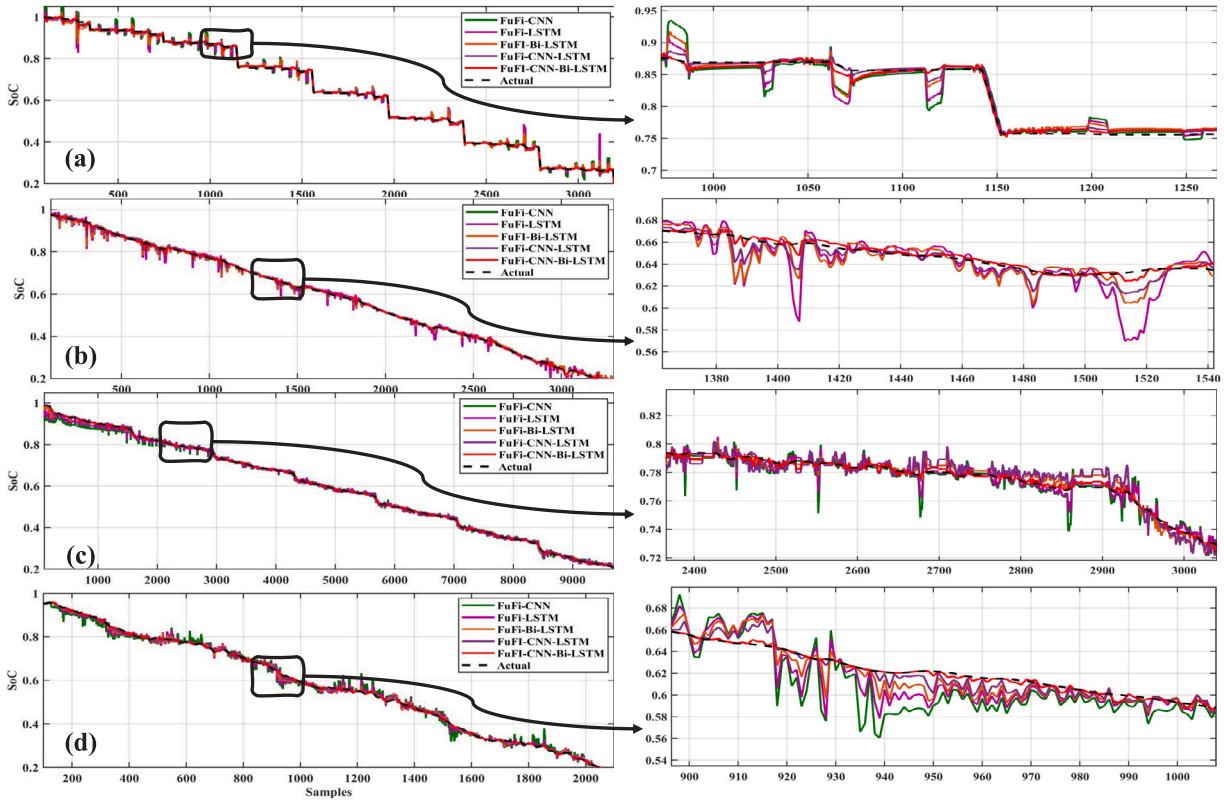

Fig. 20. SoC estimation comparison for (a) HPPC; (b) HWFET; (c) UDDS; (d) US06; at −10 degree.

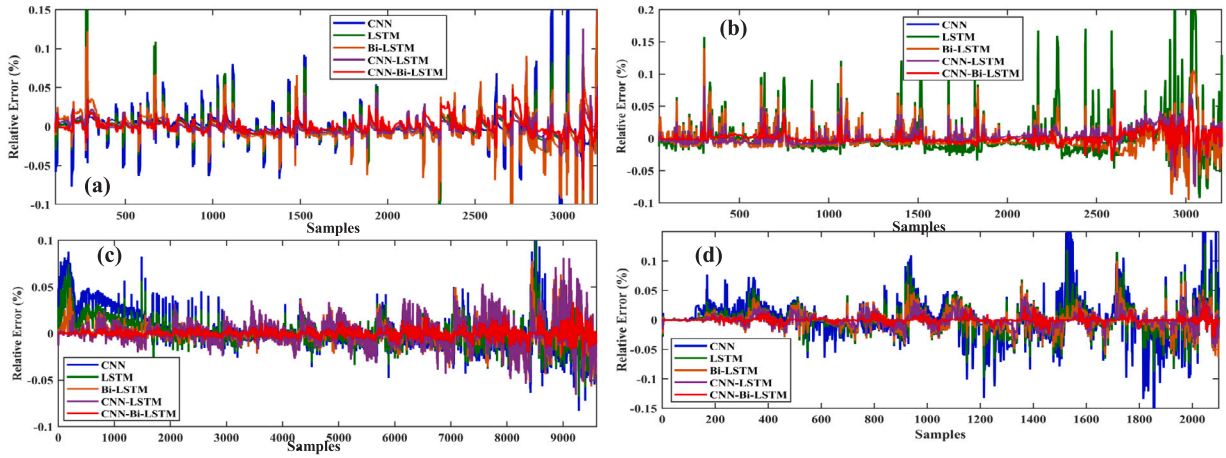

Fig. 21. Relative error comparison for (a) HPPC; (b) HWFET; (c) UDDS; (d) US06; at −10 degree.

stands as the runner-up, whereas the GWO-CNN-Bi-LSTM and PSO-CNN-Bi-LSTM exhibit the most substantial errors and the lowest R^2 values. Conclusively, the FuFi-CNN-Bi-LSTM technique demonstrates a consistently superior performance across all datasets in terms of prediction accuracy and model fit, establishing itself as the most reliable for SoC estimation, with the AOA-CNN-Bi-LSTM as the secondary preference. The GWO-CNN-Bi-LSTM and PSO-CNN-Bi-LSTM, however, may necessitate further optimization to enhance their viability for SoC estimation.

4.4. Comparative analysis with state of the art

The Table 10 provides a comparative analysis of different techniques used in high-temperature studies. The techniques evaluated include AFFRLS-RCFFUKF, Transformer+I&I, IBGRU-UKF, and FuFi-CNN-Bi-LSTM. Each technique was assessed using various datasets and temperature ranges.

Based on the reported metrics, it can be observed that the proposed FuFi-CNN-Bi-LSTM technique demonstrates superior performance compared to the other techniques. With an RMSE of only 0.05% and an NMSE of 0.06%, the FuFi-CNN-Bi-LSTM outperforms the other techniques in terms of accuracy and precision. The datasets used in this study include HPPC, HWFET, UDDS, and US06, covering temperatures of 0 °C, 10 °C, and 25 °C. This indicates the robustness of the FuFi-CNN-Bi-LSTM technique across different datasets and temperature conditions.

In comparison, the other techniques exhibit relatively higher error rates. The AFFRLS-RCFFUKF technique achieves an RMSE of 0.1% on

Table 7
Statistical analysis comparison at −10 degree temperature.

Dataset	Technique	RMSE	NMSE	MAE	RE	R^2
HPPC	FuFi-CNN-Bi-LSTM	1.1114e−03	2.0349e−03	9.0644e−03	0.00225	98.62
	FuFi-CNN-LSTM	5.4477e−01	1.5432e−01	5.7689e−02	0.0424	97.13
	FuFi-Bi-LSTM	4.2340e−01	0.207	6.4873e−01	0.1154	96.37
	FuFi-LSTM	2.6174e−01	0.283	1.5110e−02	0.8265	95.48
	FuFi-CNN	7.0085e−01	0.171	8.3808e−02	0.9795	93.66
HWFET	FuFi-CNN-Bi-LSTM	3.1558e−03	7.7259e−03	2.1460e−03	0.0501	98.47
	FuFi-CNN-LSTM	1.3768e−02	8.4253e−02	2.8695e−02	0.1283	97.40
	FuFi-Bi-LSTM	2.0331e−01	0.1801	4.4728e−02	.1568	95.92
	FuFi-LSTM	5.0038e−01	0.1401	5.6673e−01	0.1845	94.80
	FuFi-CNN	9.5088e−01	0.2899	9.0242e−01	0.2845	93.02
UDDS	FuFi-CNN-Bi-LSTM	1.3826e−03	2.7911e−023	0.0061	0.0112	98.94
	FuFi-CNN-LSTM	5.3085e−02	4.1146e−02	0.0412	0.0434	97.73
	FuFi-Bi-LSTM	2.7561e−01	2.8377e−01	0.1301	0.0975	96.07
	FuFi-LSTM	5.6396e−01	0.081	0.1948	0.1546	95.13
	FuFi-CNN	8.4971e−01	0.0133	0.5911	0.7010	93.95
US06	FuFi-CNN-Bi-LSTM	1.3411e−03	3.2620e−03	0.0021	0.0663	98.69
	FuFi-CNN-LSTM	8.3260e−02	6.0257e−02	0.061	0.1523	96.45
	FuFi-Bi-LSTM	1.2586e−01	4.6288e−01	0.0088	0.1625	95.02
	FuFi-LSTM	3.7293e−01	8.7387e−01	0.0146	0.1718	93.95
	FuFi-CNN	7.7279e−01	0.221	0.0171	0.1863	93.22

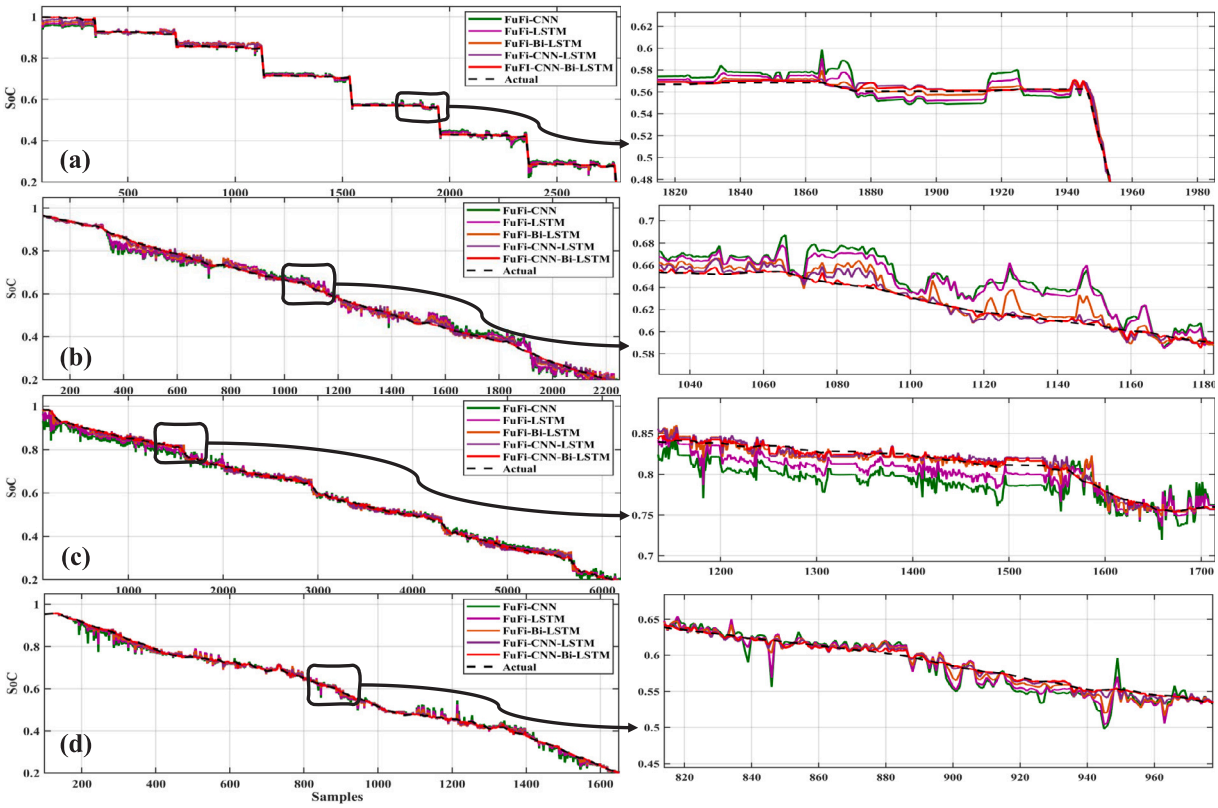

Fig. 22. SoC estimation comparison for (a) HPPC; (b) HWFET; (c) UDDS; (d) US06; at −20 degree.

the HPPC and BBDST datasets at temperatures of 15 °C, 25 °C, and 35 °C. The Transformer+I&I technique, evaluated on FUDS, US06, and DST datasets ranging from 0 °C to 50 °C, reports an RMSE of 0.54% and an MAE of 0.49%. The IBGRU-UKF technique achieves an RMSE of 0.62% and an MAE of 0.50% on the UDDS and US06 datasets at temperatures of 0 °C, 10 °C, 25 °C, and 40 °C. These results highlight the improved accuracy and performance of the FuFi-CNN-Bi-LSTM technique compared to the existing literature and state-of-the-art techniques in high-temperature analysis.

The Table 11 presents a comparative analysis of techniques used in low-temperature studies, specifically focusing on temperatures of −10 °C and −20 °C. The results demonstrate the performance of the OCV-PE technique from [43], the CNN-BWGRU technique from [5], and the FuFi-CNN-Bi-LSTM technique employed in "Our Study".

Comparing the techniques, it is evident that the proposed FuFi-CNN-Bi-LSTM technique outperforms the other techniques in terms of accuracy and precision at low temperatures. With an RMSE of only 0.11% and an NMSE of 0.10%, the FuFi-CNN-Bi-LSTM technique achieves significantly lower errors compared to the OCV-PE and CNN-BWGRU techniques. The datasets used in this study include HPPC, HWFET, UDDS, and US06, covering temperatures of −10 °C and −20 °C. The superior performance of the FuFi-CNN-Bi-LSTM technique highlights its effectiveness in accurately modeling and predicting battery behavior in low-temperature conditions.

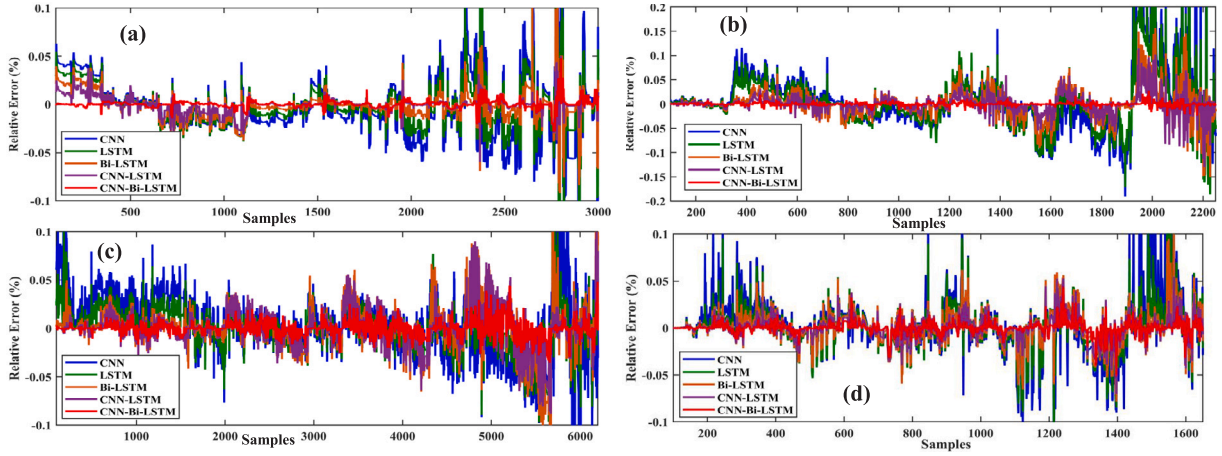

Fig. 23. Relative error comparison for (a) HPPC; (b) HWFET; (c) UDDS; (d) US06; at −20 degree.

Table 8
Statistical analysis comparison at −20 degree temperature.

Dataset	Technique	RMSE	NMSE	MAE	RE	R^2
HPPC	FuFi-CNN-Bi-LSTM	1.0342e−03	1.0144e−03	0.0021	0.0112	99.56
	FuFi-CNN-LSTM	8.0418e−02	3.5539e−02	0.045	0.0633	98.44
	FuFi-Bi-LSTM	3.6256e−01	1.6524e−01	0.0816	0.0944	97.04
	FuFi-LSTM	6.2984e−01	0.471	0.0892	0.1261	95.35
	FuFi-CNN	7.9722e−01	0.796	0.1879	0.1567	93.91
HWFET	FuFi-CNN-Bi-LSTM	1.0734e−03	2.9436e−03	0.00214	0.0078	98.94
	FuFi-CNN-LSTM	7.2265e−02	9.6878e−02	0.0563	0.0980	97.17
	FuFi-Bi-LSTM	2.1130e−01	0.0644	0.0399	0.1131	94.89
	FuFi-LSTM	6.8779e−01	0.4711	0.0782	0.1434	93.91
	FuFi-CNN	9.6591e−01	0.681	0.1199	0.1595	91.48
UDDS	FuFi-CNN-Bi-LSTM	3.3482e−03	1.0513e−03	0.0022	0.0081	98.75
	FuFi-CNN-LSTM	8.4918e−02	6.7625e−02	0.0365	0.0589	97.40
	FuFi-Bi-LSTM	1.1281e−01	0.1124	0.0841	0.0677	96.15
	FuFi-LSTM	4.1122e−01	0.1915	0.1781	0.0961	95.05
	FuFi-CNN	0.890	0.4831	0.2229	0.1151	93.97
US06	FuFi-CNN-Bi-LSTM	2.5469e−03	3.6635e−03	9.6913e−03	0.1536	98.53
	FuFi-CNN-LSTM	9.7145e−02	9.3505e−02	0.062	0.1602	96.67
	FuFi-Bi-LSTM	3.4805e−01	5.4596e−01	0.0205	0.1661	95.59
	FuFi-LSTM	8.1034e−01	0.615	0.0453	0.1737	93.26
	FuFi-CNN	9.8992e−01	0.819	0.0789	0.1855	92.15

Table 9
Comparative analysis of proposed technique with other metaheuristic algorithm based CNN-Bi-LSTM model.

Dataset	Technique	Average RMSE	Average NMSE	Average MAE	Average R^2 (%)
HPPC	FuFi-CNN-Bi-LSTM	8.99724×10^{-4}	1.008038×10^{-3}	2.156708×10^{-3}	99.154
	AOA-CNN-Bi-LSTM	1.034683×10^{-3}	1.159244×10^{-3}	24.802142×10^{-3}	97.95
	GWO-CNN-Bi-LSTM	16.898850×10^{-3}	13.331303×10^{-3}	28.522463×10^{-3}	94.22
	PSO-CNN-Bi-LSTM	24.278620×10^{-3}	15.997563×10^{-3}	34.226956×10^{-3}	92.01
HWFET	FuFi-CNN-Bi-LSTM	9.23795×10^{-4}	2.349128×10^{-3}	1.695200×10^{-3}	99.016
	AOA-CNN-Bi-LSTM	1.423640×10^{-3}	2.701497×10^{-3}	1.949480×10^{-3}	98.45
	GWO-CNN-Bi-LSTM	26.371862×10^{-3}	31.067218×10^{-3}	22.419020×10^{-3}	95.34
	PSO-CNN-Bi-LSTM	29.646235×10^{-3}	37.280661×10^{-3}	26.902824×10^{-3}	93.05
UDDS	FuFi-CNN-Bi-LSTM	8.47713×10^{-4}	4.74096×10^{-4}	5.512000×10^{-3}	99.016
	AOA-CNN-Bi-LSTM	1.698700×10^{-3}	5.452104×10^{-3}	6.338800×10^{-3}	98.59
	GWO-CNN-Bi-LSTM	19.535044×10^{-3}	62.699196×10^{-3}	72.896200×10^{-3}	95.43
	PSO-CNN-Bi-LSTM	23.442053×10^{-3}	75.239035×10^{-3}	87.475440×10^{-3}	93.87
US06	FuFi-CNN-Bi-LSTM	1.209716×10^{-3}	1.779862×10^{-3}	3.224186×10^{-3}	98.986
	AOA-CNN-Bi-LSTM	4.391173×10^{-3}	2.046841×10^{-3}	3.707814×10^{-3}	98.75
	GWO-CNN-Bi-LSTM	15.998494×10^{-3}	23.538675×10^{-3}	42.639860×10^{-3}	95.67
	PSO-CNN-Bi-LSTM	59.198193×10^{-3}	28.246410×10^{-3}	51.167832×10^{-3}	93.45

In contrast, the OCV-PE technique from [43] reports higher errors with an RMSE of 2.59% and an MAE of 4.59% on the DST dataset at −10 °C and −20 °C. Similarly, the CNN-BWGRU technique from [5] achieves an RMSE of 1.33% and an MAE of 0.99% on the LA92, UDDS, HWFET, and US06 datasets at the same temperature range. These results indicate the superior performance of the FuFi-CNN-Bi-LSTM technique in accurately predicting battery behavior at low temperatures compared to the existing literature and state-of-the-art techniques.

Table 10
Comparative analysis with existing literature and state of the art techniques for high temperatures.

Ref	Technique	Dataset	Temperature	Best error
[42]	AFFRLS-RCFFUKF	HPPC, BBDST	15 °C, 25 °C, and 35 °C	RMSE: 0.1%
[3]	Transformer+I&I	FUDS, US06 DST	0 °C, 10 °C, 20 °C, 30 °C, 40 °C, and 50 °C	RMSE: 0.54% MAE: 0.49%
[12]	IBGRU-UKF	UDDS, US06	0 °C, 10 °C, 25 °C, and 40 °C	RMSE: 0.62% MAE: 0.50%
Our study	FuFi-CNN-Bi-LSTM	HPPC, HWFET UDDS, US06	0 °C, 10 °C, and 25 °C	RMSE: 0.05% NMSE: 0.06%

Table 11
Comparative analysis with existing literature and state of the art techniques for low temperatures.

Ref	Technique	Dataset	Temperature	Best error
[43]	OCV-PE	DST	−10 °C and −20 °C	RMSE: 2.59% MAE: 4.59%
[5]	CNN-BWGRU	LA92. UDDS HWFET, US06	−10 °C and −20 °C	RMSE: 1.33% MAE: 0.99%
Our study	FuFi-CNN-Bi-LSTM	HPPC, HWFET UDDS, US06	−10 °C and −20 °C	RMSE: 0.11% NMSE: 0.10%

5. Conclusion

In conclusion, this paper presents the Fusion-Fission Optimisation (FuFi) based Convolutional Neural Network with Bi-LSTM Network (FuFi-CNN-Bi-LSTM) architecture for improving state of charge (SoC) estimation in battery systems. The proposed model combines the power of Convolutional Neural Networks (CNN) and Bi-LSTM while leveraging FuFi optimization to effectively tune the hyperparameters. The comparative analysis with other FuFi algorithm-based models highlights the superiority of the FuFi-CNN-Bi-LSTM model in terms of SoC estimation accuracy. Through rigorous testing on various drive cycle tests at temperatures ranging from −20 to 25 degrees Celsius, the FuFi-CNN-Bi-LSTM model demonstrates its robustness and reliability under different real-world operating conditions. The evaluation metrics showcase the exceptional performance of the FuFi-CNN-Bi-LSTM model. Specifically, the FuFi-CNN-Bi-LSTM model achieves an impressive RMSE of 0.05% and an NMSE of 0.06% in high-temperature scenarios, surpassing the performance of existing literature and state-of-the-art techniques. Moreover, in low-temperature settings, the model achieves an outstanding RMSE of 0.11% and an NMSE of 0.10%, outperforming other approaches in the field. These results highlight the efficacy of the proposed FuFi-CNN-Bi-LSTM model in accurately estimating the state of charge (SoC) of batteries. The proposed FuFi-CNN-Bi-LSTM model demonstrates its potential in advancing SoC estimation techniques for battery systems. The fusion of CNN and Bi-LSTM networks, along with the FuFi optimization, enhances the accuracy and reliability of SoC estimation across a wide range of temperatures, enabling more efficient battery management and improved performance in various real-world applications.

CRediT authorship contribution statement

Muhammad Hamza Zafar: Conceptualization, Data curation, Methodology, Resources. **Noman Mujeeb Khan:** Data curation, Resources, Software. **Mohamad Abou Houran:** Data curation, Investigation. **Majad Mansoor:** Validation, Visualization, Writing – original draft, Writing – review & editing. **Naureen Akhtar:** Funding acquisition, Methodology, Supervision. **Filippo Sanfilippo:** Funding acquisition, Investigation, Project administration, Resources, Supervision, Writing – original draft, Writing – review & editing.

Declaration of competing interest

Declarations of interest: None.

All authors claim that there is not any conflict of interest regarding the above submission. The work of this submission has not been published previously. It is not under consideration for publication elsewhere. Its publication is approved by all authors and that, if accepted, it will not be published elsewhere in the same form, in English or in any other language, including electronically without the written consent of the copyright-holder.

Data availability

Data used in this work is publicly available at https://calce.umd.edu/battery-data#Storage.

Acknowledgment

This research is supported by the Biomechatronics and Collaborative Robotics Group at the Top Research Center Mechatronics (TRCM), University of Agder (UiA), Norway.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.energy.2024.130584.

References

[1] Xi Z, Wang R, Fu Y, Mi C. Accurate and reliable state of charge estimation of lithium ion batteries using time-delayed recurrent neural networks through the identification of overexcited neurons. Appl Energy 2022;305:117962.
[2] Ren X, Liu S, Yu X, Dong X. A method for state-of-charge estimation of lithium-ion batteries based on PSO-LSTM. Energy 2021;234:121236.
[3] Shen H, Zhou X, Wang Z, Wang J. State of charge estimation for lithium-ion battery using Transformer with immersion and invariance adaptive observer. J Energy Storage 2022;45:103768.
[4] Liu C, Li D, Wang L, Li L, Wang K. Strong robustness and high accuracy in predicting remaining useful life of supercapacitors. APL Mater 2022;10(6):061106.
[5] Cui Z, Kang L, Li L, Wang L, Wang K. A hybrid neural network model with improved input for state of charge estimation of lithium-ion battery at low temperatures. Renew Energy 2022;198:1328–40.
[6] Qi K, Zhang W, Zhou W, Cheng J. Integrated battery power capability prediction and driving torque regulation for electric vehicles: A reduced order MPC approach. Appl Energy 2022;317:119179.
[7] Zhou W, Lu Q, Zheng Y. Review on the selection of health indicator for lithium ion batteries. Machines 2022;10(7):512.
[8] Pavlov D. H2SO4 electrolyte–an active material in the lead–acid cell. In: Lead-acid batteries: Science and technology. Amsterdam: Elsevier; 2011, p. 117–48.
[9] Feng X, Chen J, Zhang Z, Miao S, Zhu Q. State-of-charge estimation of lithium-ion battery based on clockwork recurrent neural network. Energy 2021;236:121360.
[10] Xu P, Liu B, Hu X, Ouyang T, Chen N. State-of-Charge estimation for lithium-ion batteries based on fuzzy information granulation and asymmetric Gaussian membership function. IEEE Trans Ind Electron 2021;69(7):6635–44.
[11] Padhy S, Panda S. Application of a simplified Grey Wolf optimization technique for adaptive fuzzy PID controller design for frequency regulation of a distributed power generation system. Prot Control Mod Power Syst 2021;6:1–16.

[12] Cui Z, Kang L, Li L, Wang L, Wang K. A combined state-of-charge estimation method for lithium-ion battery using an improved BGRU network and UKF. Energy 2022;259:124933.
[13] Zhengxin J, Qin S, Yujiang W, Hanlin W, Bingzhao G, Lin H. An Immune Genetic Extended Kalman Particle Filter approach on state of charge estimation for lithium-ion battery. Energy 2021;230:120805.
[14] Du B, Yu Z, Yi S, He Y, Luo Y. State-of-charge estimation for second-life lithium-ion batteries based on cell difference model and adaptive fading unscented Kalman filter algorithm. Int J Low-Carbon Technol 2021;16(3):927–39.
[15] Kong L, Liu M, Huang H, Xu Y, Bu X-H. Metal/covalent-organic framework based cathodes for metal-ion batteries. Adv Energy Mater 2022;12(4):2100172.
[16] Fang Y, Zhang Q, Zhang H, Xu W, Wang L, Shen X, Yun F, Cui Y, Wang L, Zhang X. State-of-charge estimation technique for lithium-ion batteries by means of second-order extended Kalman filter and equivalent circuit model: Great temperature robustness state-of-charge estimation. IET Power Electron 2021;14(8):1515–28.
[17] Lipu MSH, Hannan MA, Hussain A, Saad MH, Ayob A, Uddin MN. Extreme learning machine model for state-of-charge estimation of lithium-ion battery using gravitational search algorithm. IEEE Trans Ind Appl 2019;55(4):4225–34.
[18] Yang F, Li W, Li C, Miao Q. State-of-charge estimation of lithium-ion batteries based on gated recurrent neural network. Energy 2019;175:66–75.
[19] Yang F, Zhang S, Li W, Miao Q. State-of-charge estimation of lithium-ion batteries using LSTM and UKF. Energy 2020;201:117664.
[20] Deng Z, Hu X, Lin X, Che Y, Xu L, Guo W. Data-driven state of charge estimation for lithium-ion battery packs based on Gaussian process regression. Energy 2020;205:118000.
[21] Dou J, Ma H, Zhang Y, Wang S, Ye Y, Li S, Hu L. Extreme learning machine model for state-of-charge estimation of lithium-ion battery using salp swarm algorithm. J Energy Storage 2022;52:104996.
[22] Chen J, Zhang Y, Wu J, Cheng W, Zhu Q. SOC estimation for lithium-ion battery using the LSTM-RNN with extended input and constrained output. Energy 2023;262:125375.
[23] Gong D, Gao Y, Kou Y. Parameter and state of charge estimation simultaneously for lithium-ion battery based on improved open circuit voltage estimation method. Energy Technol 2021;9(9):2100235.
[24] Sun H, Yang D, Wang L, Wang K. A method for estimating the aging state of lithium-ion batteries based on a multi-linear integrated model. Int J Energy Res 2022;46(15):24091–104.
[25] Sidorov D, Panasetsky D, Tomin N, Karamov D, Zhukov A, Muftahov I, Dreglea A, Liu F, Li Y. Toward zero-emission hybrid AC/DC power systems with renewable energy sources and storages: A case study from lake Baikal region. Energies 2020;13(5):1226.
[26] Li H, Wang S, Islam M, Bobobee ED, Zou C, Fernandez C. A novel state of charge estimation method of lithium-ion batteries based on the IWOA-AdaBoost-Elman algorithm. Int J Energy Res 2022;46(4):5134–51.
[27] Mamo T, Wang F-K. Long short-term memory with attention mechanism for state of charge estimation of lithium-ion batteries. IEEE Access 2020;8:94140–51.
[28] Nouhi B, Darabi N, Sareh P, Bayazidi H, Darabi F, Talatahari S. The fusion–fission optimization (FuFiO) algorithm. Sci Rep 2022;12(1):12396.
[29] Qin J, Pan W, Xiang X, Tan Y, Hou G. A biological image classification method based on improved CNN. Ecol Inform 2020;58:101093.
[30] Wang H, Liu Z, Peng D, Qin Y. Understanding and learning discriminant features based on multiattention 1DCNN for wheelset bearing fault diagnosis. IEEE Trans Ind Inf 2019;16(9):5735–45.
[31] Agarap AF. Deep learning using rectified linear units (relu). 2018, arXiv preprint arXiv:1803.08375.
[32] Yu Y, Si X, Hu C, Zhang J. A review of recurrent neural networks: LSTM cells and network architectures. Neural Comput 2019;31(7):1235–70.
[33] Smagulova K, James AP. A survey on LSTM memristive neural network architectures and applications. Eur Phys J Spec Top 2019;228(10):2313–24.
[34] Sagheer A, Kotb M. Time series forecasting of petroleum production using deep LSTM recurrent networks. Neurocomputing 2019;323:203–13.
[35] Zou Y, Zhang Y, Yan J, Jiang X, Huang T, Fan H, Cui Z. A robust license plate recognition model based on bi-lstm. IEEE Access 2020;8:211630–41.
[36] Shrestha A, Li H, Le Kernec J, Fioranelli F. Continuous human activity classification from FMCW radar with Bi-LSTM networks. IEEE Sens J 2020;20(22):13607–19.
[37] Huang X, Li Q, Tai Y, Chen Z, Liu J, Shi J, Liu W. Time series forecasting for hourly photovoltaic power using conditional generative adversarial network and Bi-LSTM. Energy 2022;246:123403.
[38] Zha W, Liu Y, Wan Y, Luo R, Li D, Yang S, Xu Y. Forecasting monthly gas field production based on the CNN-LSTM model. Energy 2022;124889.
[39] Ali ASA, Ebrahimi S, Ashiq MM, Alasta MS, Azari B. CNN-Bi LSTM neural network for simulating groundwater level. Environ Eng 2022;8:1–7.
[40] Nguyen THT, Phan QB. Hourly day ahead wind speed forecasting based on a hybrid model of EEMD, CNN-Bi-LSTM embedded with GA optimization. Energy Rep 2022;8:53–60.
[41] Ewees AA, Al-qaness MA, Abualigah L, Abd Elaziz M. HBO-LSTM: Optimized long short term memory with heap-based optimizer for wind power forecasting. Energy Convers Manage 2022;268:116022.
[42] Feng J, Cai F, Yang J, Wang S, Huang K. An adaptive state of charge estimation method of lithium-ion battery based on residual constraint fading factor unscented Kalman filter. IEEE Access 2022;10:44549–63.
[43] Wang Q, Qi W. New SOC estimation method under multi-temperature conditions based on parametric-estimation OCV. J Power Electron 2020;20:614–23.