World Electric Vehicle Journal

MDPI

Article

An RNN-CNN-Based Parallel Hybrid Approach for Battery State of Charge (SoC) Estimation Under Various Temperatures and Discharging Cycle Considering Noisy Conditions

Md. Shahriar Nazim, Md. Minhazur Rahman, Md. Ibne Joha and Yeong Min Jang *

Department of Electronic Engineering, Kookmin University, Seoul 02707, Republic of Korea; shahriarnazim45@gmail.com (M.S.N.); minhaz.eee.97@gmail.com (M.M.R.); ibne.joha17@gmail.com (M.I.J.)
* Correspondence: yjang@kookmin.ac.kr; Tel.: +82-(2)-910-5068

Abstract: With the increasing use of lithium-ion (Li-ion) batteries in electric vehicles (EVs), accurately measuring the state of charge (SoC) has become crucial for ensuring battery reliability, performance, and safety. In addition, EVs operate in different environmental conditions with different driving styles, which also cause inaccurate SoC estimation resulting in reduced reliability and performance of battery management systems (BMSs). To address this issue, this work proposes a new hybrid method that integrates a gated recurrent unit (GRU), temporal convolution network (TCN), and attention mechanism. The TCN and GRU capture both long-term and short-term dependencies and the attention mechanism focuses on important features within input sequences, improving model efficiency. With inputs of voltage, current, and temperature, along with their moving average, the hybrid GRU-TCN-Attention (GTA) model is trained and tested in a range of operating cycles and temperatures. Performance metrics, including average RMSE (root mean squared error), MAE (mean absolute error), MaxE (maximum error), and R^2 score indicates the model is performing well, with average values of 0.512%, 0.354%, 1.98%, and 99.94%, respectively. The proposed model performs well under both high and low noise conditions, with an RMSE of less than 2.18%. The proposed hybrid approach is consistently found to be superior when compared against traditional baseline models. This work offers a potential method for accurate SoC estimation in Li-ion batteries, which has an important impact on clean energy integration and battery management systems in EVs.

Keywords: battery; EV; GRU; Huber loss; Li-ion battery; state of charge estimation; SoC; TCN

Citation: Nazim, M.S.; Rahman, M.M.; Joha, M.I.; Jang, Y.M. An RNN-CNN-Based Parallel Hybrid Approach for Battery State of Charge (SoC) Estimation Under Various Temperatures and Discharging Cycle Considering Noisy Conditions. *World Electr. Veh. J.* **2024**, *15*, 562. https://doi.org/10.3390/wevj15120562

Academic Editor: Haijun Ruan

Received: 8 November 2024
Revised: 1 December 2024
Accepted: 3 December 2024
Published: 4 December 2024

1. Introduction

The growing market share of battery electric vehicles (EVs) and the increasing demand for renewable energies have highlighted the importance of efficient and reliable battery technologies [1,2]. EVs' energy storage systems are increasingly relying on lithium-ion (Li-ion) batteries as their primary power source, given their exceptional attributes such as high energy density, high power density, long lifecycle, and environmentally sustainable characteristics [3,4]. Developing accurate methods for estimating the state of charge (SoC) of EV batteries is crucial for maximizing their performance, lifespan, and overall efficiency [5,6]. The BMS is essential to monitor the state of the battery based on real-time data and protect the battery from overcharge and over-discharge to ensure the safety and reliability of EV operation [6,7]. In this rapidly growing field, there is a need for advanced techniques to precisely measure the status of the battery. The complexities of battery chemistry, temperature, and aging make this a complex and challenging task [8]. Moreover, the accurate estimation of SoC is essential for optimizing the deployment of renewable energies and enhancing grid stability [9]. Battery SoC means how long the battery can last without being recharged [10,11]. SoC can only be evaluated indirectly using factors like current. voltage and temperature [12]. Up till now, there have been five general methodologies for estimating SoC: look-up table methods, Coulomb counting, model-based

approaches, data-driven approaches, and hybrid approaches [2,5,13,14]. However, those models can only work under strict conditions, and the traditional machine learning-based model cannot capture the sequence properly. That is why it is important to adopt new advanced technology and approaches to estimate SoC properly.

Individual SoC estimation methods can be classified into four categories based on the battery model and the estimation algorithms: ampere-hour counting, model-based approaches, filter-based techniques, and data-driven methods. By integrating current over time, the ampere-hour counting approach determines SoC variation based on the assumption of a precise initial SoC [2,15]. Ampere-hour counting for battery SoC estimation relies on an accurate initial SoC assumption, leading to cumulative errors and the inability to consider self-discharge [16]. As an open-loop method, this approach may experience limited accuracy in current sampling, leading to a degradation of estimates over time. Zheng et al. [17] show that ampere-hour counting gives around 5% current drift of 5 mA over 100 h.

Model-based methods are widely preferred for SoC estimation as they use voltage signals to establish a relationship between voltage and SoC. Among these, the open-circuit voltage (OCV) is a reliable indicator used in various modeling techniques such as electrochemical models (EMs) [18], equivalent circuit models (ECMs), and fractional-order models (FOMs) [19]. OCV curves are often employed to provide an initial estimate for parameter tuning in model-based approaches. Electrochemical models, which simulate the internal mechanisms of the battery, are used for SoC estimation through a method known as electrochemical model-based SoC estimation [20]. The estimation of battery SoC using electrochemical models remains challenging due to computational complexity, the need for precise parameterization, and difficulties in adjusting to actual operating circumstances and battery aging dynamics [21]. ECMs are simplified electrical representations used to estimate battery SoC based on voltage and current measurements, offering simplicity and computational efficiency compared with other models [22]. ECMs frequently face problems in representing transient and nonlinear behavior because of their sensitivity to accurate parameter variations [19]. Zheng et al. [17] found an overall 5% SoC estimation error for 1 mV yield in voltage simulation. Fractional-order models provide improved accuracy for battery SoC estimation, incorporating complex dynamics such as the Butler–Volmer equation and fractional calculus, crucial for electric vehicle management systems [23]. Fractional-order models for battery SoC estimation may suffer from increased voltage prediction error during large current pulses and computational burden, limiting their effectiveness in real-time applications [24]. Overall, model-based methods for battery SoC estimation can be sensitive to parameter variations and may require accurate knowledge of the battery's characteristics [25].

Filter-based methods for battery SoC estimation employ algorithms to enhance performance [13,26]. Popular methods include Kalman filters [27], extended Kalman filters [28,29], the cubature Kalman filter [30], and particle filters [31], providing efficient and robust SoC predictions. Filter-based methods for battery SoC estimation can be computationally intensive and complex, requiring careful tuning of parameters for optimal performance [15]. Li et al. [32] show a comparison between three different types of state-of-the-art filter-based approaches and found that SoC error can reach up to 7.3%.

As artificial intelligence has developed, a significant amount of study has been carried out on data-driven methods for analyzing the SoC of batteries [33]. Data-driven methods, including machine learning techniques, offer promising avenues for battery SoC estimation by utilizing large datasets. These approaches include various strategies such as Gaussian process regression (GPR), support vector machines (SVMs), recurrent neural networks (RNNs), and convolutional neural networks (CNNs), each suited to different types of data and modeling requirements [34–37]. GPR offers a robust framework for battery SoC estimation, providing a reliable estimation of uncertainty and enhancing the accuracy up to 95%, increasing SoC predictions' reliability. However, GPR for battery SoC estimation may encounter scalability and computational complexity issues due to kernel function selection

and hyperparameter optimization [38]. The SVM offers a robust and adaptable solution for accurately estimating battery SoC in real-world conditions, making it a valuable and reliable strategy for electric vehicle battery management systems [39]. Anton et al. [40] proposed SVM-based SoC estimation approaches with support vectors and found that mean absolute error (MAE) is less than 4% but the maximum error is up to 6% due to its limitation of handling nonlinearity. Nevertheless, SVMs for battery SoC estimation may suffer from limitations in handling nonlinearities and may require significant computational resources for training [41].

CNN models offer promising solutions for battery SoC estimation by utilizing their ability to capture spatial dependencies in battery data, enhancing accuracy and robustness in prediction tasks [42]. The 1D-CNN effectively captures Li-ion battery data features but shows lower SoC estimation precision than other neural network structures. Bhattacharjee et al. [43] used the 1D-CNN-based model to estimate battery SoC and showed that the MAE of estimation is around 4.72%. Meanwhile, temporal convolutional networks (TCNs), designed for time series data, exhibit lower SoC estimation reliability than other methods [44]. Hannan et al. [45] proposed a multi-layer time-domain convolution layer based on TCNs, which achieved a MSE of around 0.85%. On the other hand, RNNs, including architectures like long short-term memory (LSTM) and the gated recurrent unit (GRU), have demonstrated effectiveness in battery SoC estimation by capturing long-term dependencies and nonlinear relationships in battery data [33,46]. However, LSTM models for battery SoC estimation may face challenges with overfitting when trained with numerous input parameters, potentially compromising generalization ability in real-world scenarios [47]. Using GRU model parameters reduces and increases prediction accuracy, which leads to reducing overfitting issues, but cannot fully solve the vanishing gradient issue when the model parameters are large [48]. Xiao et al. [49] proposed a GRU-based model and found that it can achieve a MAE of up to 0.49% for the FUDS driving cycle. Nowadays, some research is being conducted in the area of advanced algorithms like transfer learning and hybrid models for SoC estimation. Bain et al. [50] proposed an LSTM encoder–decoder-based hybrid model that can capture the long-term sequence and estimate future directions. The proposed model can estimate battery SoC with an MAE value of 1.07%. On the other hand, hybrid models like CNN-LSTM for battery SoC estimation have proven useful in measuring and precisely capturing nonlinear relationships among factors such as voltage, current, and temperature, consistently providing effective performance [46]. Song et al. [9] proposed a CNN-LSTM model with a root mean squared error (RMSE) of 1.31%. A TCN-LSTM SoC estimation model was proposed by Hu et al. [51] with a MAE and RMSE of 0.70% and 0.60%, respectively. In addition, the TCN-LSTM model combines the robustness of the TCN with the ability of long short-term memory (LSTM) to capture long-term dependence in input data and shows considerable promise for accurately estimating battery SoC [52]. A CNN-GRU-based model was proposed by Huang et al. [53] where CNN and GRU models are connected in series, with an RMSE of 1.54% and MAE of 1.26%. Li et al. [54] used the TCN, GRU, and attention in series with moving averages to estimate SoC more accurately. In their study, they found that the model is capable of SoC estimation with an RMSE of 0.96% and an MAE of 0.80%.

The SoC estimation relies on various measured parameters, including voltage, current, and temperature. These measurements are inherently susceptible to noise, manifesting as both bias and variance in sensor data, which can significantly impact estimation accuracy. While high-quality sensors and proper shielding can minimize such noise in controlled laboratory environments, real-world applications present numerous challenges. These include electromagnetic interference, suboptimal data acquisition algorithms, environmental factors, and improper sensor calibration [55]. Although existing research has proposed algorithms for noise detection and elimination [56], few AI models can effectively process noisy data without preliminary noise filtering techniques. Moreover, existing AI models often struggle to maintain consistent performance with acceptable error margins in SoC estimation under different operational conditions. To address these limitations, this study

presents a novel GRU-TCN-Attention (GTA) model for improving battery SoC estimation accuracy. The model integrates the TCN and GRU to capture sequential patterns, while the attention mechanism between layers enhances the model's sequence learning capabilities. Additionally, the implementation of moving averages for feature processing enables more effective capture of long-term trends and helps in noise immunity. The model's robustness is validated through comprehensive testing under various drive cycles, temperature conditions, and noisy conditions, with performance benchmarked against attention-based TCN (A-TCN) and attention-based GRU (A-GRU) models. The important contributions are:

1. The proposed architecture combines the TCN and GRU to effectively capture long sequence dependencies in battery SoC data. Integrating these models enhances temporal flexibility, reduces overfitting, and improves noise resistance. Adding self-attention between the GRU and TCN layers further refines the model, allowing it to focus on key input features for better feature extraction. Finally, it also takes advantage of both CNN and RNN feature extraction processes and automated noise-resilient capabilities.
2. The Huber loss function is used to ensure stable model training by avoiding the transient and noise impacts of voltage and current during model training. To capture the overall trends, the Huber loss function is utilized in conjunction with the moving average of voltage, current, and temperature. This procedure improves the model's ability to track long-term trends as well as protect it from noise, anomalies, and transients.
3. The model's performance is examined under a variety of temperature conditions and driving cycles to prove its generalization capabilities. Furthermore, a comparison with A-TCN and A-GRU models demonstrates how the proposed technique maintains and improves SoC estimation accuracy by utilizing TCN performance while leveraging the GRU's advantages in capturing lengthy sequences of SoC data.
4. Proposed model performance is tested under different noisy conditions to ensure that the model is noise resistant.

The rest of the paper is structured as follows. Section 2 focuses on the construction of the model and the processing of datasets to efficiently capture the SoC. Section 3 discusses the findings of this study and compares the model's performance with other models and recent studies. The conclusion of this study is drawn in Section 4.

2. Methodology

2.1. Dataset Description

For a data-driven approach to the proposed method and the related complexity and uncertainty of the loading condition of Li-ion batteries, a publicly available LG18650 Li-ion battery dataset, provided by McMaster University, is used for training the model [57]. In this dataset, a brand-new LG18650 Li-ion battery is tested in a controlled thermal chamber with a 75 amp, 5 V universal battery tester channel with voltage and current measuring error rates less than 0.1% [58]. The dataset records the variation of voltage, current, temperature, and capacitance of the battery in electric vehicles at temperatures varying from −20 °C to 40 °C, with a data resolution of 0.01 s. The dataset includes common operating conditions such as Mixed 1-8, Urban Dynamometer Driving Schedule (UDDS), LA92 Dynamometer Driving Schedule (LA92), Highway Fuel Economy Driving Schedule (HWFET), and US06 Supplemental FTP Driving Schedule (US06).

Different charging and discharging driving cycles are included in the test from these driving cycles at different temperatures. From these driving cycles, UDDS, LA92, HWFET, and US06 are considered for model testing at temperatures including 0 °C, 10 °C, and 25 °C, and under the same temperature conditions Mixed 1-8 is used for training the model. From the training dataset, 10% of the data is used for validation at the time of training. LA92 represents the urban driving conditions in Los Angeles, which typically consist of moderate speeds with frequent stops and starts. Therefore, the current and voltage profile contains frequent fluctuation due to sudden acceleration and deceleration. On the other

hand, HWFET represents highway driving conditions where speed is kept steady for a long period, which causes the voltage and current to become more stable compared with others. At the same time, UDDS represents urban driving conditions where speed is kept less steady compared with HWFET as there are repeated cycles of acceleration, steady-state cruising, and deceleration, which are also reflected in the current and voltage profile. Lastly, US06 represents an aggressive driving style where very fast acceleration and deceleration are happening and the fluctuation of the voltage and current profile is comparatively high and frequent. Battery main specifications are described in Table 1. The training process was conducted on a custom-built desktop PC equipped with an AMD Ryzen 5 5600X, 128 GB of DDR4 RAM operated in 3200 MHz, powered by an NVIDIA GeForce RTX 4090 graphics processing unit with 24 GB of GDDR6X memory.

Table 1. Battery specifications.

Parameters	Value
Chemistry	Li[NiMnCo]O_2 (H-NMC)/Graphite + SiO
Nominal voltage	3.6 V
Discharge	2 V (end voltage), 20 A (max continuous current)
Nominal capacity	3 Ah
Nominal discharge current	3 A

2.2. Noise Addition and Data Preprocessing

Systematic noise injection was implemented in the training data to evaluate and enhance the model's resilience to sensor noise, incorporating both bias and variance components. Although sensor bias typically exhibits temporal variation in practical applications, this study employs a constant bias assumption for simplicity. The noise model is expressed through Equation (1), where ΔV, ΔI, and ΔT represent the constant bias terms for voltage, current, and temperature measurements, respectively. At the same time, δV, δI, and δT denote their corresponding random noise components. The random noise components are modeled as independent and identically distributed (i.i.d.) variables with zero mean and constant variance (σ), as defined in Equation (2). Figure 1 shows the impact of noise addition in the battery measured data and Table 2 contains the parameters that are used to add noise in the training data.

$$\begin{aligned} V_{t,m} &= V_{t,true} - \Delta V - \delta V_t \\ I_{t,m} &= I_{t,true} - \Delta I - \delta I_t \\ T_{t,m} &= T_{t,true} - \Delta T - \delta T_t \end{aligned} \tag{1}$$

$$\begin{aligned} E(\delta V) = 0, \sigma^2(\delta V) &= \sigma_V^2 \\ E(\delta I) = 0, \sigma^2(\delta I) &= \sigma_I^2 \\ E(\delta T) = 0, \sigma^2(\delta T) &= \sigma_T^2 \end{aligned} \tag{2}$$

To ensure a comprehensive capture of the dataset's long-term trends, primary features such as instantaneous voltage, current, and temperature are utilized, alongside their respective moving averages. This moving average calculation is crucial for capturing long-term trends and immunity from transient values, using Equation (3), where y_{ma} represents the moving average and y_1, y_2, y_3 denote individual entries, with a fixed window size of 600. The selected features are then normalized using the min-max normalization approach, as described in Equation (4), where x represents the original value and $x_{\min}$ and $x_{\max}$ indicate the minimum and maximum values for the relevant feature. The scaling ranges between 0 and 1. This normalization procedure has the advantage of minimizing the impact

of outliers. After that, data are segmented into fixed window sizes of 75, which are then allocated into train, validation, and test datasets.

$$y_{ma} = \frac{y_1 + y_2 + y_3 + ... + y_n}{n} \tag{3}$$

$$x_n = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{4}$$

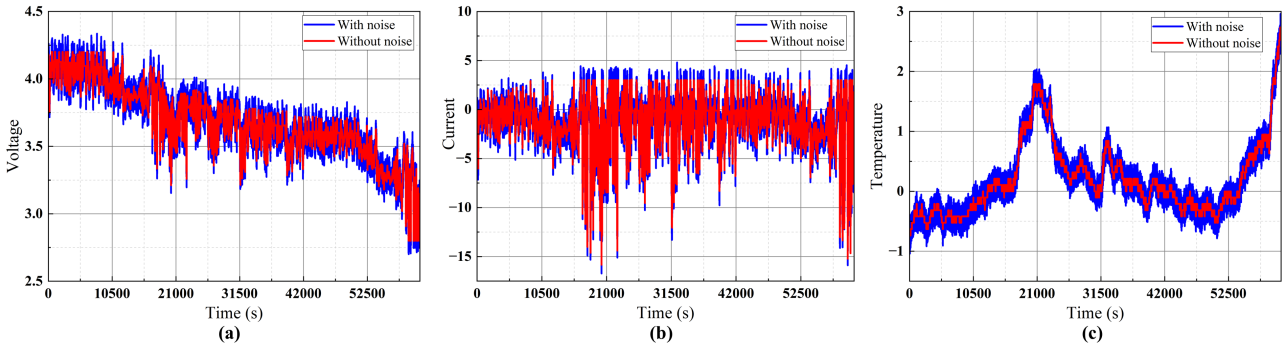


Figure 1. Impact of noise addition on (**a**) voltage, (**b**) current, and (**c**) temperature at 0 °C.

Table 2. Noise parameters.

Parameter	Value
ΔV	0.1 V
ΔI	0.3 A
ΔT	0.5 °C
σ_V^2	0.04
σ_I^2	0.5
σ_T^2	0.09

2.3. Model Description

In the proposed architecture, the model is a combination of a TCN and GRU with an attention mechanism, which is used for sequence capture while keeping focus on important sequence elements. The model consists of three TCN blocks and three GRU blocks, with an attention block placed between each pair. Following that, the TCN and GRU outputs are concatenated and passed through a fully connected dense layer. Table 3 provides a full model description, and Figure 2 depicts the architectural plan.

Table 3. Hyper-parameters of the proposed model.

Parameters	Values
Kernel size	9
No. of filters	64
Learning rate	0.001–0.00001
Loss function	Huber
Optimizer	Adam
Training epochs	300
Window size	75

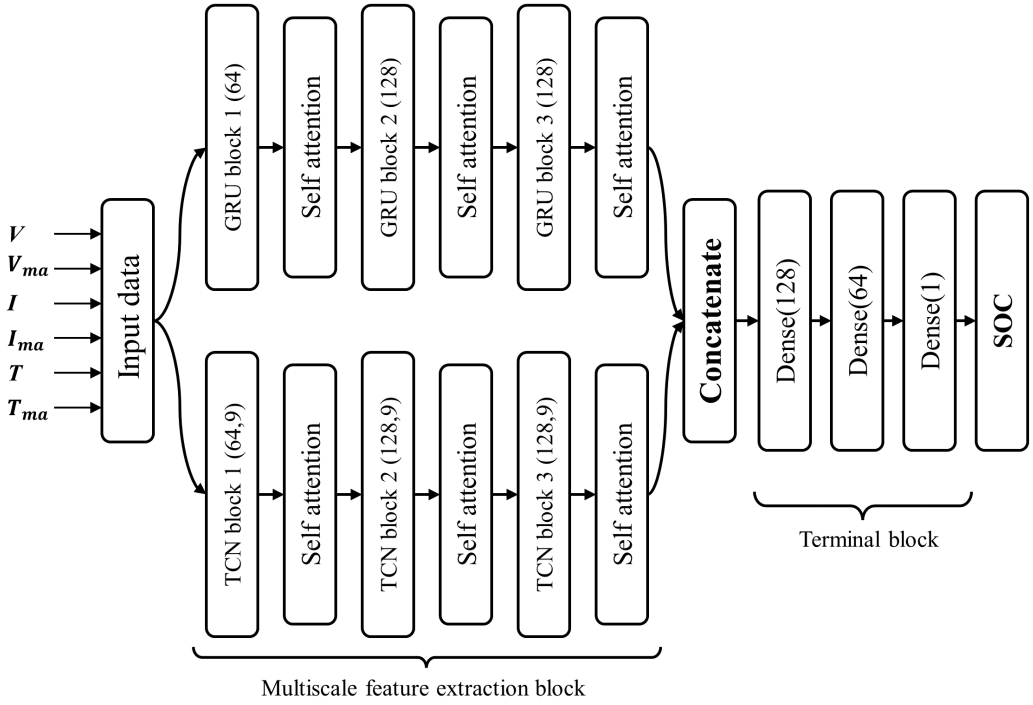


Figure 2. Proposed architecture.

2.3.1. Attention Temporal Convolution Network (A-TCN)

Though the TCN is used for image classification [59], it can be used for sequence modeling with the help of causal convolution, dilated convolution, and residual connection [60,61]. It provides parallel computation, low memory consumption, and flexible perceptual field size, making it more robust than other methods [62]. The TCN consists of three main parts: causal convolution, dilation, and the residual connection between input and output. In causal convolution, the output only depends on the current and past elements of the input, which prevent data leakage from future value, and the dilation helps the model increase the receptive field exponentially without increasing the depth of the model, which helps the model capture the sequence more effectively. The residual connection allows layers to learn modification to the identity mapping rather than the entire transformation, which helps the network be more stable.

The fundamental layer of the TCN is the convolution layer, but it uses a spatial convolution named dilated causal convolution. In this work, the causal convolution layer contains three convolution layers, which are shown in Figure 3. The main difference between causal convolution and traditional convolution is that input is calculated using only the past and present values of the previous layer, which prevents data leakage from future values. That means the output of time t is calculated using the elements from the $t-1$ and t of the previous layer, which ensures that the model does not violate the natural temporal order by peaking into the future. With the help of unidirectional data flow and causal connections, the TCN becomes a strictly time-dependent model. For a causal layer with a filter size k, input sequence $x(n)$, i-th with output $y(n)$ define as:

$$y(n) = \sum_{k=1}^{K} x(k)h(n-k) \tag{5}$$

The filter output at each given time (n) depends solely on the current and previous input values ($x(k)$ for $k \leq n$). This is because the summation only takes into account shifted filter values ($h(n-k)$) for $k \leq n$. Future input values ($k > n$) do not affect the current output.

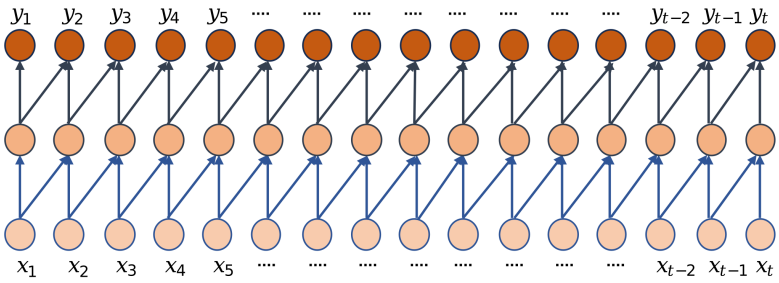


Figure 3. Causal convolution.

By using only causal convolution it is hard to capture sequences, and to resolve this issue dilation is used. In the TCN, sequence dilation alleviates this constraint and ensures that sequential dependencies are captured completely. Dilation includes increasing the space between filter elements after each layer, which expands the network's receptive field. A dilation factor, indicated as d, increases the space between filter elements by d times, hence expanding the receptive field by d times. This deliberate use of dilation allows TCNs to record greater temporal contexts and dependencies across numerous time steps, hence improving the network's capacity to extract meaningful features and patterns from sequential data. If the dilation factor is d then output $y(n)$ is -

$$y_n = \sum_{i=0}^{k-1} g(i) * x_{t-d \cdot i} \tag{6}$$

In the proposed model, the dilation rate is set as 2^n, where n ranges from 0 to 2, with $g(i)$ denoting the i-th element of the filter. The dilated causal and only dilated convolution is shown in Figure 4. However, as the number of layers increases during training, neural networks often encounter the vanishing gradient effect. To mitigate this issue, skip connections or residual connections are introduced in the TCN network [63]. These connections ensure the efficient transfer of information across layers. Specifically, the output of the TCN layer after a residual connection is defined as -

$$y_n = A(g(x) + ConvD(x)) \tag{7}$$

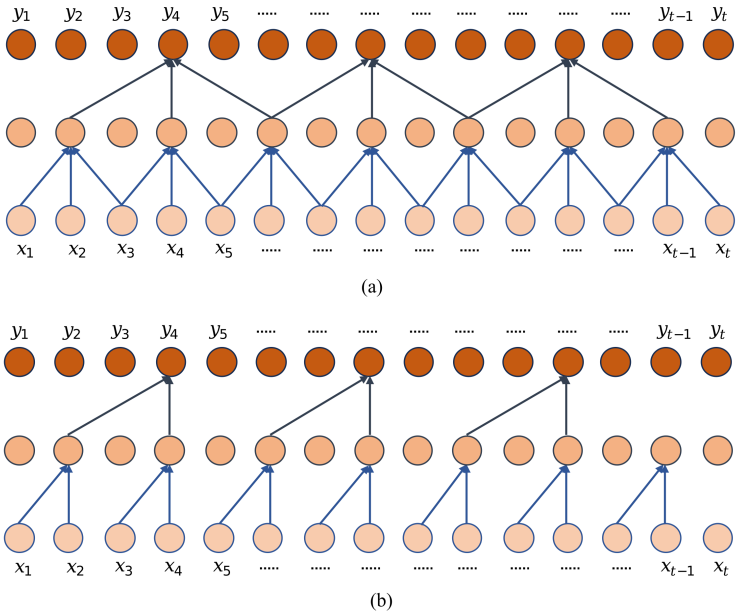


Figure 4. (**a**) Dilated convolution, (**b**) dilated causal convolution.

In this configuration, y_n denotes the final output of the TCN block, where x represents the input to the TCN block. The outputs of $g(x)$ and $ConvD(x)$ are concatenated and subsequently passed through an activation function, A, specifically *ReLU* in this instance. Figure 5a illustrates a single TCN block, while Figure 5b showcases the overall TCN architecture with an incorporated attention mechanism.

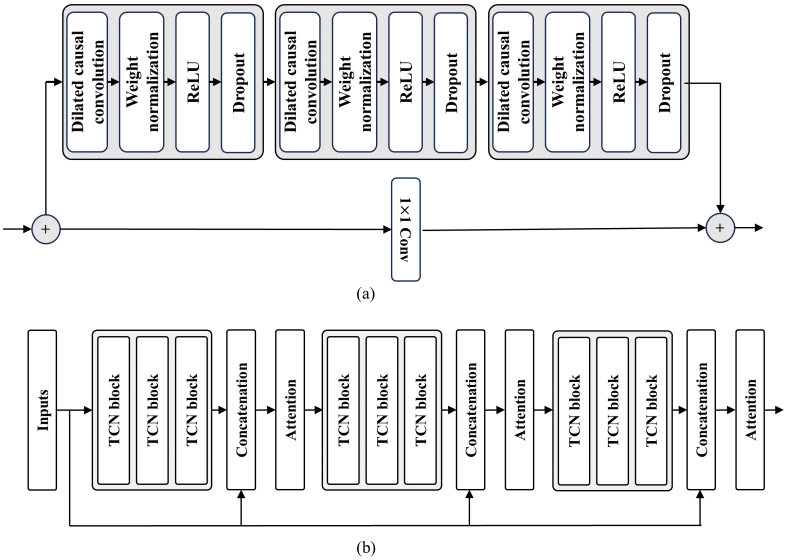


Figure 5. (**a**) Single TCN block, (**b**) TCN arrangement.

2.3.2. Attention Gated Recurrent Unit (A-GRU)

In terms of capturing extended sequences, the gated recurrent unit (GRU) outperforms classic recurrent neural network (RNN) and long short-term memory (LSTM) models [64]. Additionally, the GRU has benefits that include faster training with fewer parameters, a simpler structure, and adaptability to the vanishing or exploding gradient problem [65]. The GRU uses two gating methods when estimating the SoC: the update gate and the reset gate. The update gate controls the flow of information from the previous hidden state, determining whether data are maintained and passed on in subsequent cycles by using Equation (8). Meanwhile, the reset gate determines how much past information is needed to forget and creates a new vector for the candidate hidden state using Equation (9). To generate the new candidate hidden state vector in Equation (10), a hyperbolic tangent (tanh) activation function is applied to the input information vector and the reset gate output. The vector from the update gate and the candidate hidden state vector are then combined to produce the new hidden state vector using Equation (11). The model architecture is shown in Figure 6.

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \tag{8}$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \tag{9}$$

$$\tilde{h}_t = \tanh(W_h \cdot [r_t \odot h_{t-1}, x_t]) \tag{10}$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{11}$$

where W_z, W_r, and W_h are the weights for the update gate, reset gate, and candidate hidden state, respectively. Whereas, h_{t-1} is the output of the previous GRU unit and x_t is the input at time t.

2.3.3. Attention

The attention mechanism is a special type of sequence-to-sequence modeling task that translates a source sequence into a target sequence. It is inspired by the human vision attention by which it finds the most important features of the input sequence [66]. Compared with traditional sequence modeling where the entire sequence is converted into

a fixed-sized context vector, attention creates a new vector where each feature is weighted dynamically depending on its importance [67]. The basic attention architecture is shown in Figure 7. The hidden state output (h_t) and the optimized output of the attention layer are calculated as -

$$u_t = \tanh(W_h h_t + b) \tag{12}$$

$$\alpha_t = \text{softmax}(v^T u_t) \tag{13}$$

$$y_t = \sum_{t=1}^{M} \alpha_t h_t \tag{14}$$

where W_h ($W_h \in \mathbb{R}^{K_w \times K_w}$) is defined as the weight matrix, b ($b \in \mathbb{R}^{K_w}$) is the biased term of the attention mechanism, and α_t is the probability distribution [68]. The attention mechanism significantly enhances feature extraction by focusing on the most relevant aspects of the input data and capturing long-range dependencies critical for SoC estimation. By computing weighted representations, it prioritizes important features such as the relationships between input values, while filtering out noise and redundancy. Integrated after each GRU and TCN block, the attention layers refine temporal and hierarchical features, enabling a multiscale representation with improved contextual representation. This dynamic adjustment of feature importance ensures an effective fusion of outputs, enhancing the model's robustness and accuracy across varying operating conditions.

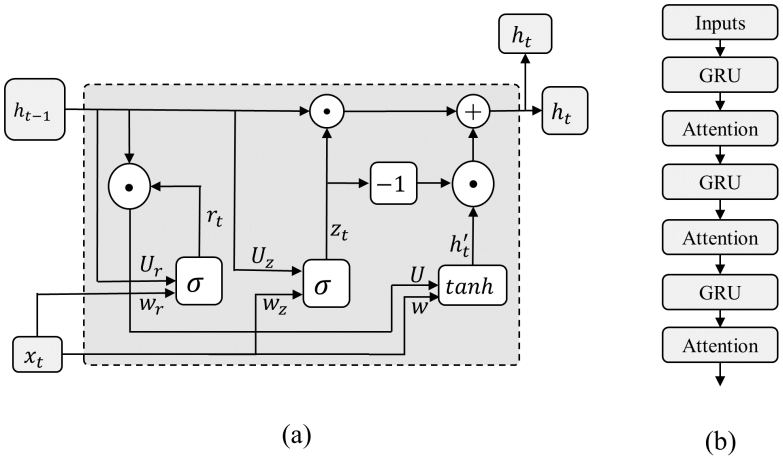


Figure 6. (**a**) Single GRU block, (**b**) GRU block with attention.

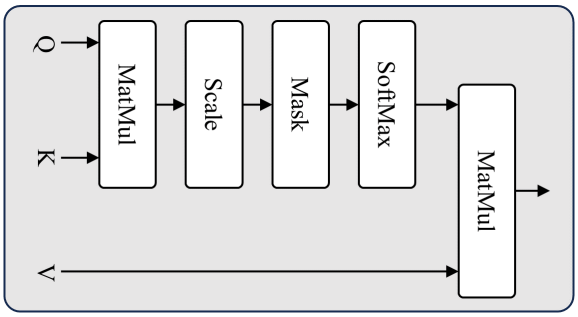


Figure 7. Attention mechanism.

2.4. Overall Model Architecture

To accurately estimate battery SoC under varying operating conditions and sensor noise, this work proposes a parallel hybrid combination-based model that combines the feature extraction capabilities of RNNs and CNNs with an attention mechanism to enhance the identification of critical features. The model architecture, illustrated in Figure 2, consists

of two main components: the multiscale feature extraction block and the terminal block. The multiscale feature extraction block utilizes the TCN and GRU to capture both long-term and short-term dependencies and effectively utilizes the strengths of RNNs and CNNs to improve accuracy. A self-attention mechanism is integrated between the TCN and GRU layers to prioritize and refine the most relevant patterns, further enhancing the feature extraction process. The terminal block plays a crucial role in mapping the rich, high-dimensional features extracted by the multiscale feature extraction block to the final SoC estimation output. This block is designed to simplify and interpret the learned features into a comprehensible and actionable output by employing a fully connected dense layer. The dense layer aggregates and processes the input features, ensuring the model retains the most critical information for SoC estimation while discarding irrelevant details. To improve robustness, this block includes dropout layers to reduce overfitting, as well as an activation function to introduce nonlinearity and ensure the model adapts effectively to complex patterns in battery behavior. By focusing on efficient feature mapping and maintaining computational efficiency, the terminal block ensures the final SoC predictions are both accurate and reliable. This seamless integration of feature extraction and mapping enhances the overall robustness of the proposed model.

2.5. Huber Loss Function

The Huber loss function is is robust loss function for regression tasks that is resilient to outliers or sensor noise and ensures stable model training [69]. It combines the advantages of MSE and MAE by adjusting tunable parameters called delta (δ) [70]. The process is a piecewise function that switches operating conditions based on error rate and can be expressed by Equation (15), where α ($y_{true} - y_{pred}$) represents the error rate and $L_\delta(\alpha)$ is the loss. When α is less than δ it acts like the MSE and greater than δ it acts like the MAE. On the other hand, for small δ it is less sensitive to large errors that are penalized linearly rather than quadratically and more sensitive to outliers where the loss is higher because more residuals fall within the quadratic region. For this study, δ is chosen as 1.75. This value is found using grid search techniques where different δ values are tested and the δ value of 1.75 is found to be the most optimal value.

$$L_\delta(a) = \begin{cases} \frac{1}{2}a^2 & \text{for } |a| \le \delta, \\ \delta \cdot \left(|a| - \frac{1}{2}\delta\right) & \text{for } |a| > \delta, \end{cases} \tag{15}$$

2.6. Evaluation Metrics

Three distinct evaluation metrics are used to assess the performance of the proposed approach. Those are MAE, RMSE, and maximum error (MaxE). MAE indicates how accurately the model is estimating the value compared with the real value, RMSE indicates the robustness of the model, and MaxE indicates the largest error model provided during the testing period. Finally, the R^2 score is evaluated on how well the values predicted by the model fit with the actual value by using Equation (19). For the total n number of entities, the formulas of MAE, RMSE, MaxE, and R^2 score are defined as -

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{16}$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{17}$$

$$\text{MaxE} = \max_{i=1}^{n}|y_i - \hat{y}_i| \tag{18}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \bar{y})^2}{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{19}$$

where y_i is the true SoC value from the experiments, $\hat{y}_i$ is the SoC value estimated by the proposed model, and $\bar{y}$ is the mean of the true SoC.

3. Results and Discussion

With the rising adoption of EVs and battery storage systems, accurate estimation of battery SoC becomes increasingly critical. To address this challenge, this study proposes a hybrid RNN and CNN-based approach, utilizing the TCN and GRU with inter-layer attention mechanisms. The model's ability is evaluated across diverse drive cycles and temperature conditions, followed by a comparative analysis with attention-based TCN and GRU models. In addition, model performance is also evaluated under different noisy conditions. Subsequently, the proposed model's performance is compared with recent studies, showing its better accuracy and efficiency. As the effect of battery internal resistance is varied by SoC level, for better analysis of the proposed model performance overall the SoC curve is divided into three parts: region 1 (1 to 0.8), region 2 (0.8 to 0.2), and region 3 (0.2 to 0).

3.1. Model Performance Under Noise-Free Condition

The overall performance of the model is evaluated across four distinct driving cycles: LA92, UDDS, US06, and HWFET, and under three different temperature conditions (0 °C, 10 °C, and 25 °C) without any additional noise. The evaluation results for the proposed model across the LA92, US06, UDDS, and HWFET driving cycles are illustrated in Figures 8–11 and the statistical analysis is shown in Table 4.

Table 4. Proposed model performance evaluation.

Drive Cycle	Temperature (°C)	RMSE (%)	MAE (%)	MaxE (%)	R^2 Score (%)
LA92	0	0.612	0.418	4.7	99.95
	10	0.29	0.24	1.3	99.98
	25	0.45	0.36	1.75	99.97
US06	0	0.28	0.22	1.06	99.98
	10	0.26	0.21	0.9	99.99
	25	0.89	0.77	2.8	99.88
UDDS	0	0.74	0.47	5.54	99.92
	10	0.55	0.4	2.98	99.96
	25	0.33	0.27	0.8	99.98
HWFET	0	1.08	0.85	4.25	99.84
	10	0.16	0.11	0.4	99.99
Avg.	-	0.512	0.354	1.98	99.94

For the HWFET drive cycle, characterized by steady and high-speed conditions leading to rapid battery SoC depletion, the model demonstrates optimal performance due to fewer fluctuations in current and voltage. Consequently, there are minimal current and voltage transients, enabling the model to operate efficiently, shown in Figure 8. Notably, at 10 °C, the model achieves its best results, boasting the lowest RMSE, MAE, MaxE, and R^2 scores of 0.16%, 0.11%, 0.4%, and 99.99%, respectively, as depicted in Figure 11. Conversely, the LA92 drive cycle exhibits frequent current fluctuations, with the moving average aiding the model in efficiently capturing these sequences. Across varying temperature conditions (0 °C, 10 °C, and 25 °C), the model maintains an average error below 2%, with very few higher error rates, as illustrated in Figure 9. Notably, at 0 °C, the MaxE, RMSE, and MAE reach 4.7%, 6.12%, and 0.418%, respectively, while, at 10 °C, the model achieves its best performance with a RMSE, MAE, and MaxE of 0.29%, 0.24%, and 1.3%, respectively, alongside a maximum R^2 score of 99.98%.

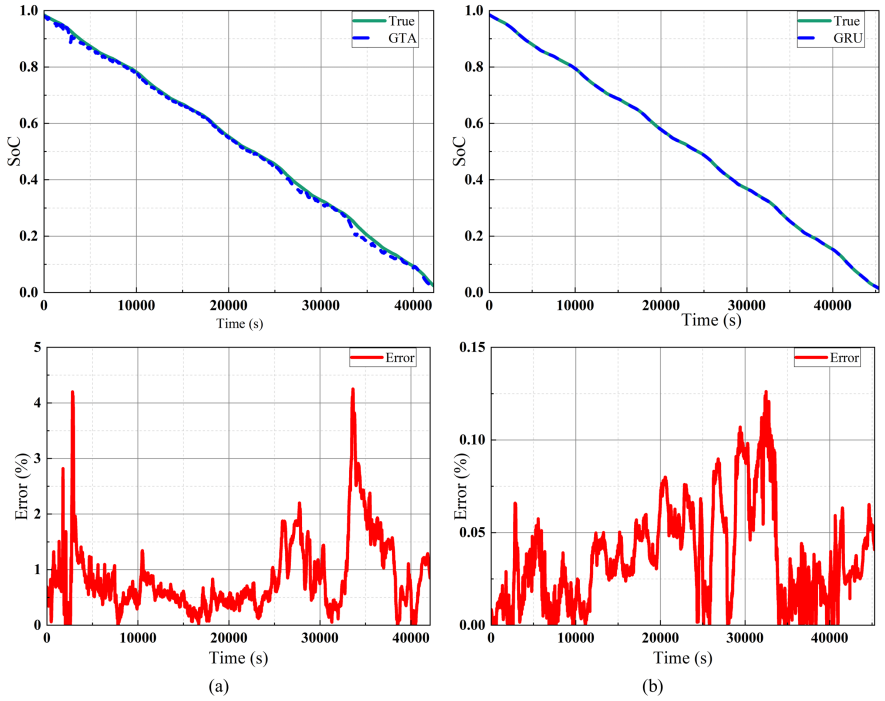


Figure 8. Performance of model for drive cycle HWFET at (**a**) 0° C, (**b**) 10 °C.

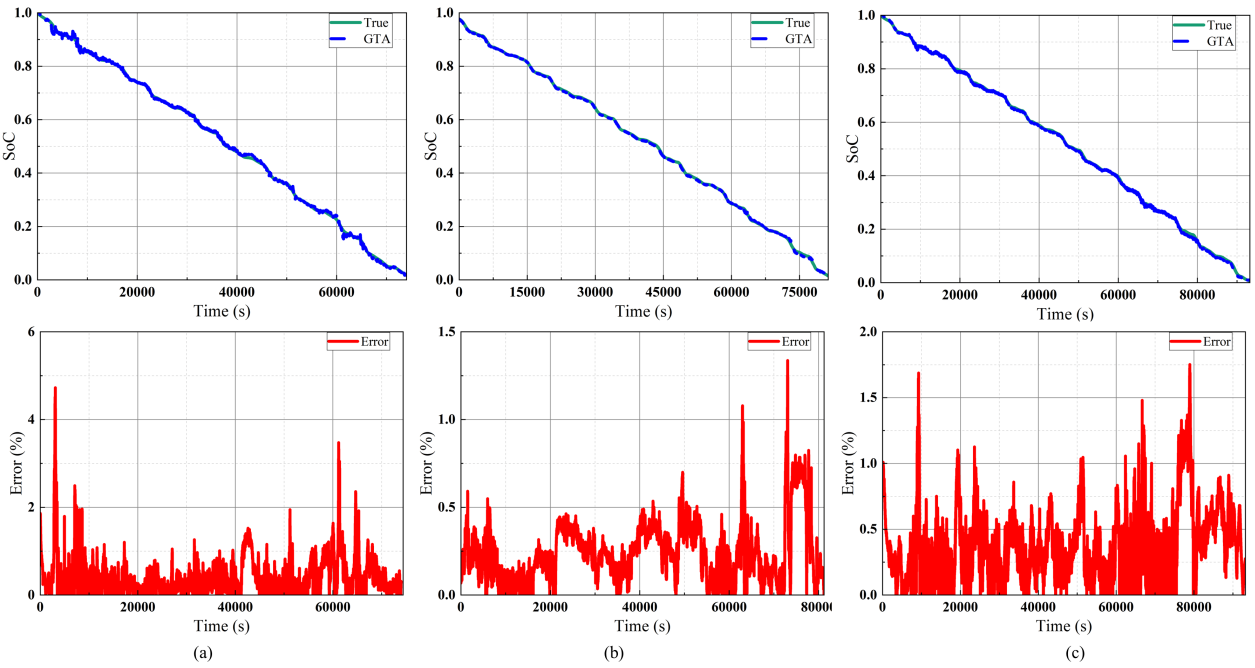


Figure 9. Performance of model for drive cycle LA92 at (**a**) 0 °C, (**b**) 10 °C, and (**c**) 25 °C.

In drive cycle UDDS, speed is kept less steady compared with HWFET, which causes frequent changes in current with transients. So, the model is facing some difficulties compared with other drive cycles. A maximum MaxE of 5.54% is found in this drive cycle. In terms of temperature conditions, the best result is found for 25 °C temperature, with RMSE, MAE, MaxE, and R^2 scores of 0.33%, 0.27%, 0.8%, and 99.98%, respectively. In the case of the US06 drive cycle, over the different temperature conditions the model performs well compared with LA92 and UDDS, which is shown in Figure 10. The best performance is found at the temperature of 10 °C, with RMSE, MAE, MaxE, and R^2 scores of 0.26%, 0.21%, 0.9%, and 99.99%, respectively.

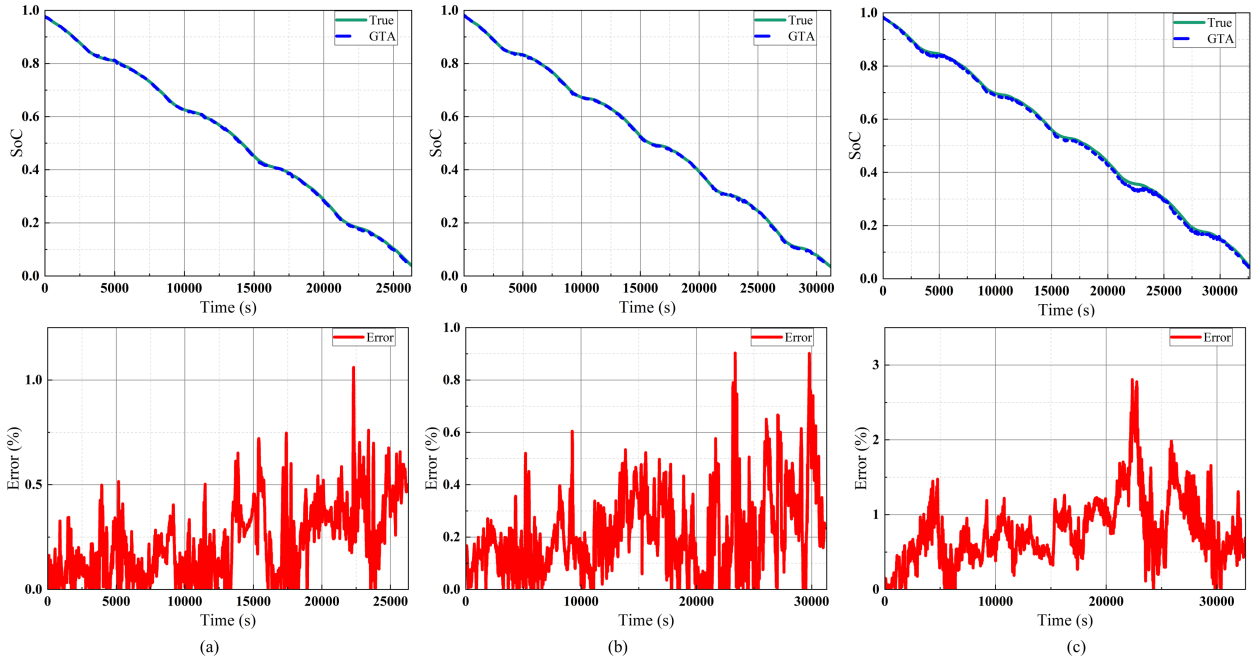


Figure 10. Performance of model for drive cycle US06 at (**a**) 0 °C, (**b**) 10 °C, and (**c**) 25 °C.

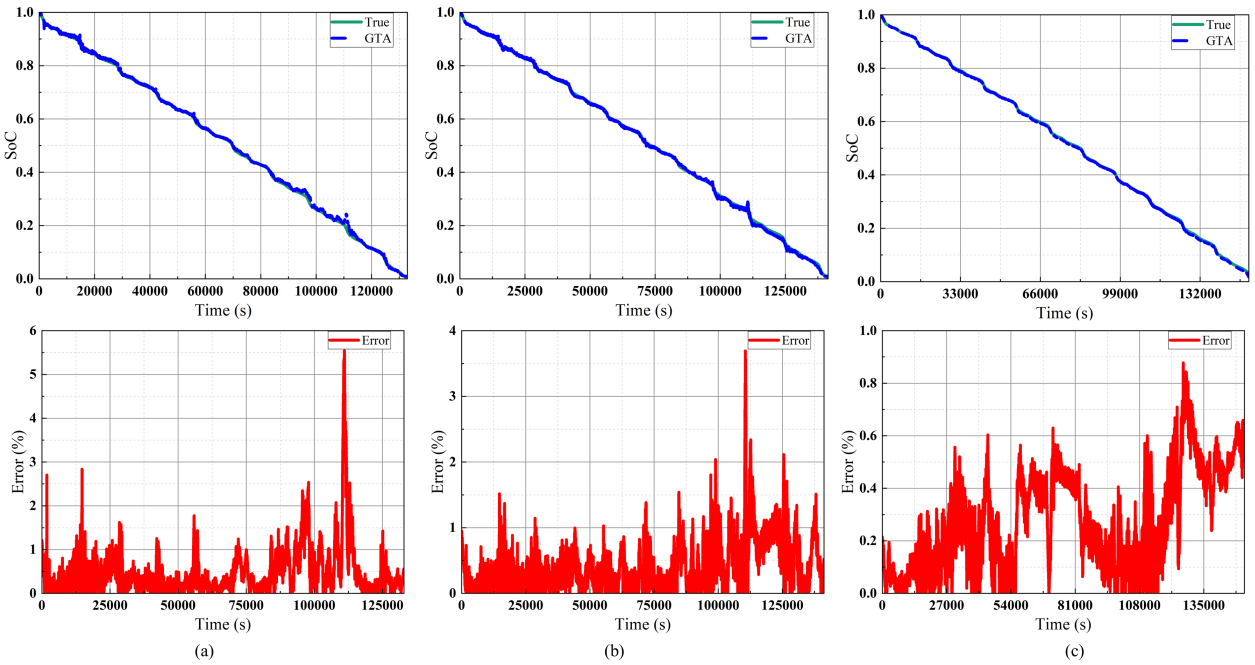
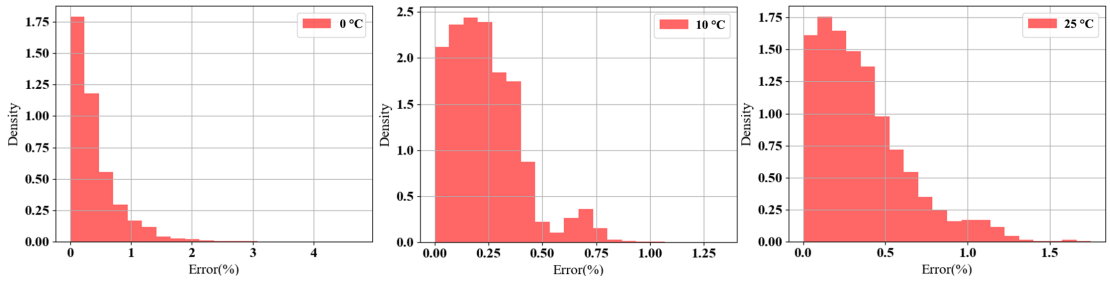


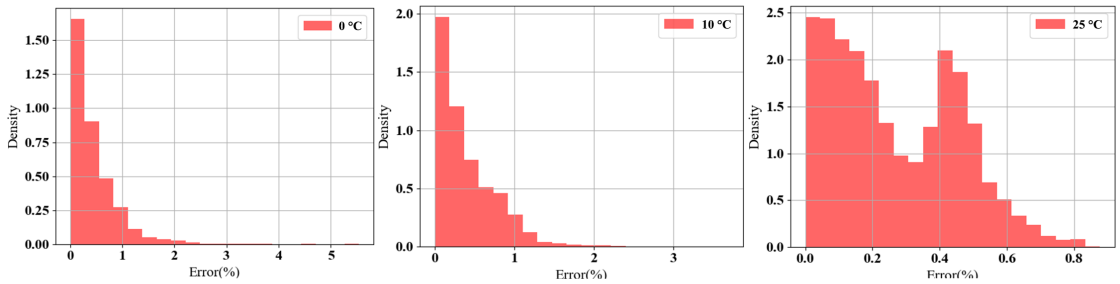
Figure 11. Performance of model for drive cycle UDDS at (**a**) 0 °C, (**b**) 10 °C, and (**c**) 25 °C.

From Figures 8–11, it is evident that the model's error rate varies across different SoC levels, with notable deviations in region 3, where the highest errors are observed. These errors primarily stem from changes in the battery's internal resistance, which significantly impact voltage and current measurements. Across various drive cycles, region 3 consistently exhibits the highest error rates, with the MaxE often occurring in this region. In addition, this results in higher deviation in this region compared with other regions. In region 2, the error rate is generally stable, with only occasional spikes. However, an exception is noted in the HWFET drive cycle at 10 °C, where frequent fluctuations are observed, deviating from the otherwise consistent performance in this region. In region 1, the error rates are moderate but occasionally feature large spikes. For instance, significant deviations are observed in the LA92 drive cycle at 0 °C and 25 °C, as well as in the HWFET drive cycle

at 0 °C. These variations highlight the sensitivity of the model's performance to changes in internal resistance, particularly at lower SoC levels. Since internal resistance strongly depends on the battery's SoC, its variation disproportionately affects the model's accuracy at low SoC levels compared with higher SoC levels, where the error rates tend to be lower and more consistent. Figures 12 and 13 present the error distributions for SoC estimation under different drive cycles and temperature conditions, showcasing the performance and robustness of the proposed method. For the US06 drive cycle (Figure 13a), the error distribution at 0 °C and 10 °C is highly concentrated below 0.3%, indicating remarkable accuracy, even at lower temperatures. At 25 °C, the error spread increases slightly, but most errors remain below 1%, demonstrating the model's consistent performance at higher temperatures. Similarly, for the HWFET drive cycle (Figure 13b), the errors at 0 °C are primarily concentrated below 1%, with a few outliers extending up to 4%. At 10 °C, the error distribution becomes even more compact, with most values below 0.1%, highlighting the model's exceptional precision under moderate conditions. For the LA92 drive cycle (Figure 12a), the error distributions across 0 °C, 10 °C, and 25 °C remain tightly packed, with the majority of errors falling below 1%. While the 0 °C condition shows slightly higher deviations, the overall accuracy remains commendable. The UDDS drive cycle (Figure 12b) exhibits a slightly more dispersed error pattern at 0 °C, with errors extending up to 2%. However, at 10 °C and 25 °C, the errors are concentrated below 1%, reflecting the model's ability to maintain consistent performance across varying conditions.



(a) Drive cycle LA92

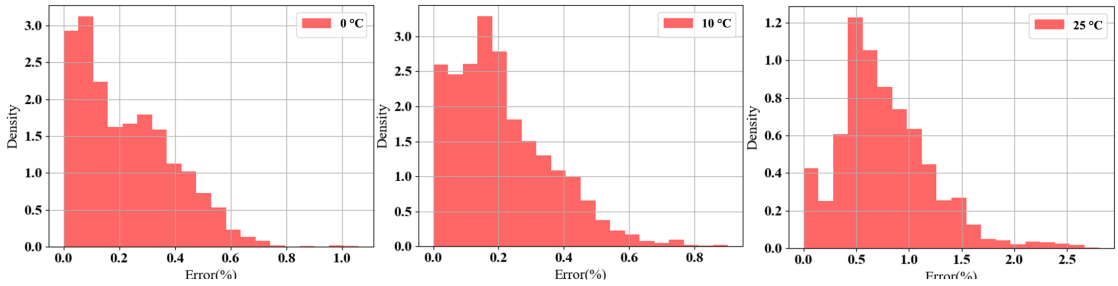


(b) Drive cycle UDDS

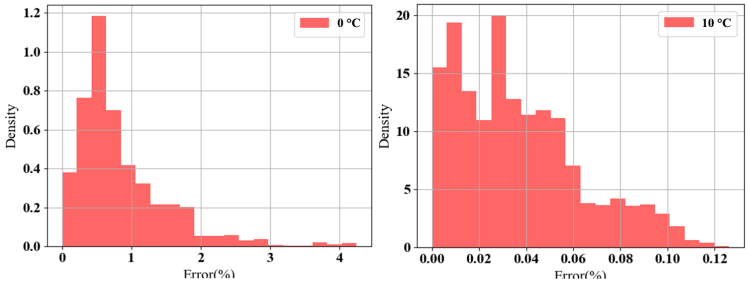
Figure 12. SoC estimation error (%) distribution for drive cycles (**a**) LA92 and (**b**) UDDS.

In Table 4, the overall model performance is shown in terms of different drive cycles and temperature conditions. The average model performance in terms of RMSE, MAE, MaxE, and R^2 scores of 0.512%, 0.354%, 1.98%, and 99.94%, respectively, shows that the proposed model can estimate SoC with high accuracy and more efficiently. For the temperature condition of 0 °C, the proposed model performance is found with larger RMSE, MAE, MaxE, and lower R^2 scores than other temperature conditions. For almost all the drive cycles, its error rate is found to be high, except with US06. In the case of the US06 drive cycle, the error rates are found in between the maximum and minimum values. This indicates that the model has some limitations in the low-temperature region. Regarding the temperature level of 10 °C, the proposed model performs best, based on low RMSE, MAE, MaxE, and high R^2 scores compared with other temperature conditions. An exception is found in the drive cycle of UDDS; in this case, the error rate and R^2 score are found

in between the maximum and minimum values. So, the model is most efficient at this temperature level. At the temperature level of 25 °C, the model shows mixed performance. For LA92, model performance is in between the maximum and minimum values, for drive cycle US06, its performance is the least, and, for drive cycle UDDS, its performance is the best. So, the model performance is said to be average for this temperature level. Those findings indicate that the model is optimized for 10 °C, model performance is moderate at 25 °C, and model performance is lowest at 0 °C. Though there are some limitations in region 3, the model can capture both short-term and long-term sequences with low error rates. In this approach, the TCN helps the model to capture the long dependencies using dilation, whereas the GRU enhances the capturing capability of short-term dependencies using its gating mechanism, which allows it to dynamically control the information, making it responsive to immediate changes in the sequence. In addition, this use of the attention mechanism helps the model to focus on the important data points, improving the accuracy. Those capabilities enhance the model sequence prediction capabilities in different time spans, which leads to more accurate and stable estimations. On the other hand, while the TCN excels in capturing long-term sequences and providing a stable foundation for overall trends, the GRU's ability to adjust its memory based on recent data allows for fine-tuning predictions in shorter time spans. This combination enhances the model's generalization and robustness, reducing overfitting and improving its performance across a variety of scenarios. In addition, this use of moving averages helps the model to become immune from the transient effect of voltage and current.



(a) Drive cycle US06



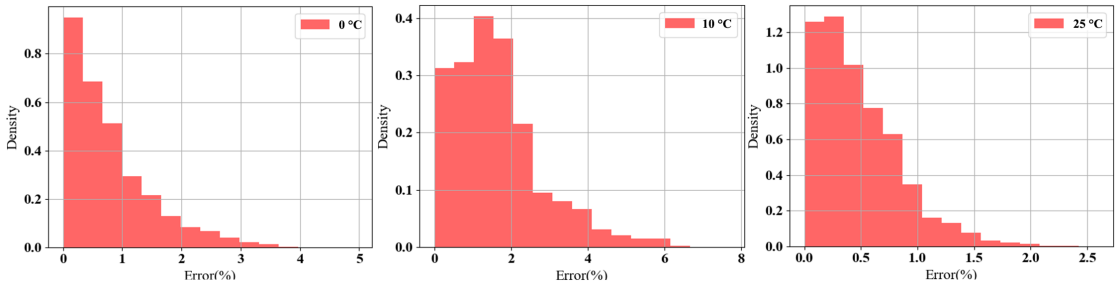
(b) Drive cycle HWFET

Figure 13. SoC estimation error (%) distribution for drive cycles (**a**) US06 and (**b**) HWFET.

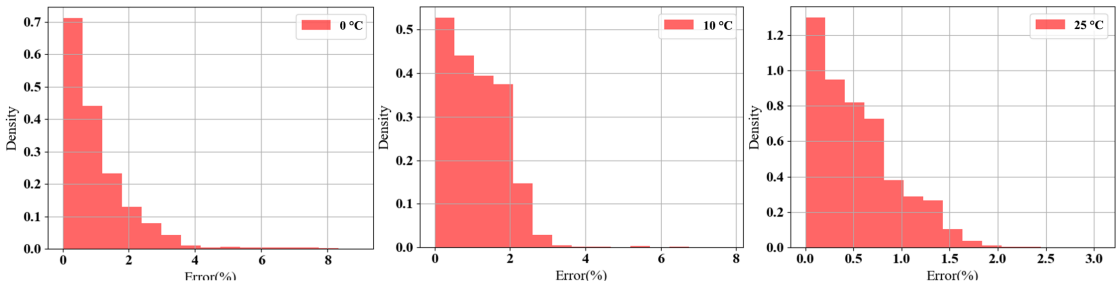
3.2. Model Performance Under Noisy Conditions

As the model performance can be affected by sensor noises, in this section the performance is evaluated under varying noise conditions to ensure its robustness and applicability in real-world scenarios. Two distinct noise environments are considered: low noise conditions with a signal-to-noise ratio (SNR) ranging from 50 dB to 60 dB, and high noise conditions with an SNR range of 20 dB to 30 dB. To test model performance in noisy conditions, RMSE is used as the evaluation metric due to its sensitivity to larger errors, providing a clear indication of how well the proposed method minimizes deviations from the true values under noisy conditions. The statistical comparison of model performance under these conditions is presented in Table 5.

The results demonstrate that the model maintains a stable performance and exhibits a low RMSE across different temperatures and drive cycles, even in the presence of noise. For low noise conditions, the estimation error is consistently below 1.5%, with average errors of 1.056%, 0.74%, and 0.78% for temperatures of 0 °C, 10 °C, and 25 °C, respectively. Similarly, under high noise conditions, the estimation error remains below 2.18%, with average errors of 1.68%, 1.59%, and 1.44% for the respective temperatures. Figures 14 and 15 showcase the SoC estimation error distributions across four standard drive cycles—US06, HWFET, LA92, and UDDS—under varying ambient temperatures (0 °C, 10 °C, and 25 °C). In urban scenarios like the LA92 cycle, 85% of errors at 0 °C are below 1%, while warmer conditions (25 °C) push this to 95%, with minimal errors exceeding 1%. The UDDS cycle, simulating stop-and-go driving, shows consistent results, with over 80% of errors below 2% at 0 °C and more than 90% below 1.5% at 25 °C. For the US06 cycle, at 0 °C, over 80% of errors are below 1%, with a sharp peak around 0.5%. At 10 °C, 70% of errors are within 2%, while at 25 °C the accuracy improves further, with nearly 90% of errors confined below 1%. Similarly, in the HWFET cycle, representing highway driving, more than 85% of errors at 0 °C remain below 2%, while at 25 °C nearly all errors are under 1.5%, with a peak near 0.3%. This statistical error distribution shows that noisy conditions result in slightly higher variability, particularly at lower temperatures, but remain within acceptable limits.



(a) Drive cycle LA92



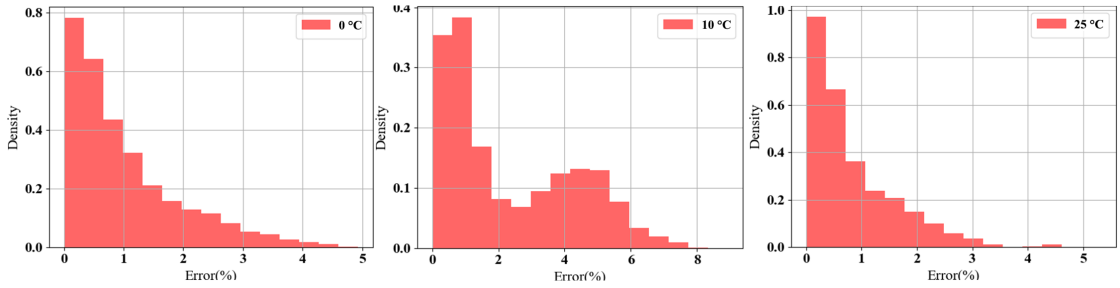
(b) Drive cycle UDDS

Figure 14. SoC estimation error (%) distribution for drive cycles (**a**) LA92 and (**b**) UDDS under high noise conditions.

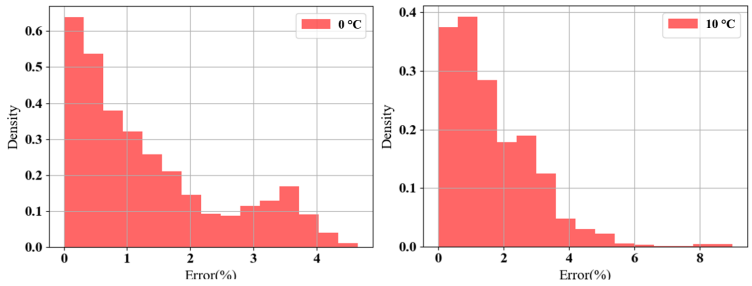
Notably, the model's noise resilience improves with an increase in temperature, as observed from the reduced RMSE values. This indicates that the system becomes less sensitive to noise at higher operating temperatures, enhancing its reliability. Additionally, the incorporation of techniques such as the moving average and the Huber loss function plays a pivotal role in improving the model's robustness. These techniques help mitigate the impact of noise by smoothing out fluctuations and reducing the influence of outliers in the data. The model's ability to limit the estimation error to such low levels, even under high noise conditions, underscores its suitability for practical applications. In real-world battery management systems, where measurement noise is inevitable, the proposed model demonstrates significant potential to deliver accurate and reliable predictions.

Table 5. Model performance under noisy conditions (RMSE (%)).

Temperature (°C)	Drive Cycle	Noise Free	Low Noise	High Noise
0	LA92	0.612	0.985	1.56
	US06	0.28	0.619	1.01
	UDDS	0.74	1.11	1.98
	HWFET	1.08	1.52	2.18
10	LA92	0.29	0.67	1.36
	US06	0.26	0.75	1.77
	UDDS	0.55	1.04	2.01
	HWFET	0.16	0.51	1.24
25	LA92	0.45	0.68	1.47
	US06	0.89	1.07	1.64
	UDDS	0.33	0.614	1.21



(a) Drive cycle US06



(b) Drive cycle HWFET

Figure 15. SoC estimation noise distribution for drive cycles (**a**) US06 and (**b**) HWFET under high noise conditions.

3.3. Model Performance Comparison

To investigate the proposed model's effectiveness in battery SoC estimation, its performance is compared with other RNN and CNN-based models, including A-TCN and A-GRU. Both A-TCN and A-GRU models contain three hidden layers and are fed with similar data. The performance, stability, and error rates of these models are compared against the proposed model to evaluate its effectiveness.

For the LA92 drive cycle, all models can follow the downward trends with oscillations, as shown in Figure 16. The A-GRU model exhibits a large deviation at 0 °C, with deviations decreasing as the temperature increases. The A-TCN model follows trends with less deviation but experiences frequent oscillations. The proposed GTA model performs the best among the three, demonstrating minimal deviation and oscillation. The GTA model effectively captures the sequence with high accuracy and stability. In region 1, both the A-TCN and A-GRU models show oscillations, while the GTA model maintains stability.

In regions 2 and 3, the A-GRU model shows high deviation and oscillation, whereas the GTA model exhibits less oscillation and deviation. Although the A-TCN model shows less deviation, its oscillation is higher compared with the GTA model. Overall, for the LA92 drive cycle, all models exhibit higher error rates at low temperatures and lower error rates at higher temperatures. However, the GTA model stands out for its more stable and accurate performance.

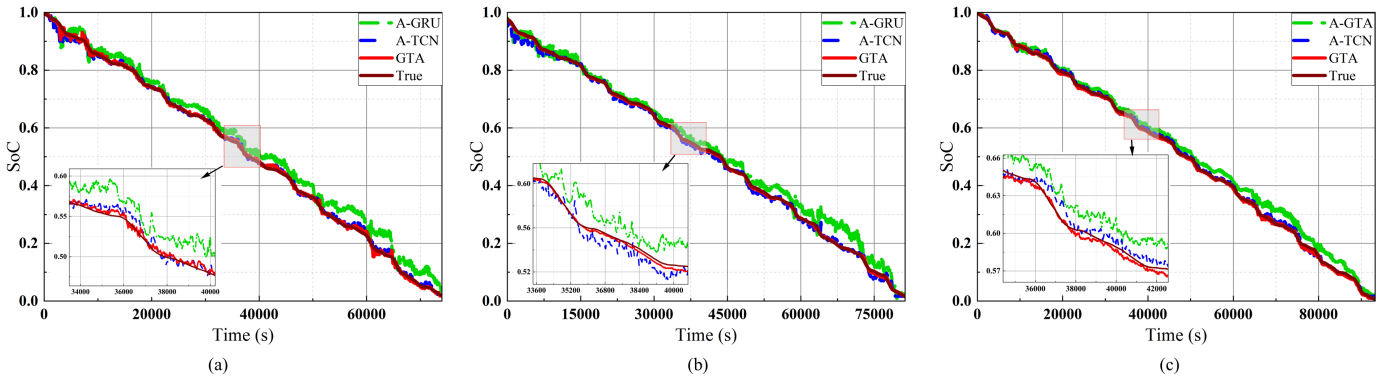


Figure 16. Model performance comparisons with A-TCN and A-GRU for drive cycle LA92 at (**a**) 0 °C, (**b**) 10 °C, and (**c**) 25 °C.

In the case of the US06 drive cycle, all three models perform well with minimal oscillation and deviation. As depicted in Figure 17, the A-GRU model exhibits higher oscillation across all temperature conditions compared with the other models. However, there is a slight deviation from the true value at the end of regions 2 and 3. While the A-TCN model displays less oscillation than the A-GRU model, it also experiences deviation in region 1 at 0 °C and 25 °C. Conversely, across all three temperature conditions, the GTA model demonstrates the highest stability and lowest error across regions 1, 2, and 3. On the contrary, for the UDDS drive cycle, the A-GRU model performs the worst, exhibiting significant deviations across regions 1, 2, and 3. Although the A-TCN model shows less deviation, it also presents higher oscillation rates than the GTA model. Figure 18 illustrates the GTA model's exceptional stability, with minimal oscillations. Notably, all models exhibit their best performance for the HWFET drive cycle, as shown in Figure 19. However, both the A-TCN and A-GRU models demonstrate frequent oscillations in regions 1 and 3. Additionally, a minor deviation is observed in region 3 at a temperature of 10 °C for the GRU model. Conversely, the GTA model consistently delivers optimal performance across both temperature conditions. In most scenarios, the A-TCN and A-GRU models exhibit some deviation and oscillations in regions 1 and 3, whereas the GTA model consistently provides stable and low-error results across all regions. This underscores the better stability of the proposed model compared with other models.

In Table 6, it is found that the GTA model performs better compared with the A-TCN and A-GRU models under each temperature condition. At the temperature of 0 °C, the best RMSE, MAE, and MaxE are 0.28%, 0.22%, and 1.06% for drive cycle US06, whereas, for the same conditions, A-TCNs and A-GRUs (RMSE, MAE, and MaxE) are 1.48% and 1.82%, 1.16% and 1.3%, and 4.11% and 7.23%, respectively. So, the GTA model performs better under this temperature condition. Similarly, the highest error rates found for the GTA models are an RMSE, MAE, and MaxE of 1.08%, 0.85%, and 4.25%, respectively. This shows that the proposed model's performance is better than the A-TCN and A-GRU models at a temperature of 0 °C. In the same way, the GTA model's lowest RMSE is found to be 0.16%, whereas the best RMSE values for A-TCN and A-GRU are 0.21% and 0.2%, respectively. If we consider the highest RMSE value for the GTA, A-TCN, and A-GRU models, it is found to be 0.55%, 1.46%, and 3.6%, respectively. For temperature condition 25 °C, the lowest value of the RMSE is 0.33% for the GTA model, 1.59% for the A-GRU model, and 0.73% for the A-TCN model. The highest RMSE values are 1.59%, 2.43%, and 1.2% for the GTA,

A-GRU, and A-TCN, respectively. So, under all three temperature conditions, the proposed model outperforms the A-GRU and A-TCN models, which demonstrates that the proposed model is better than the other two models. Table 7 shows the comparison of the model performance with the baseline machine learning model with the statistical models. The table compares the performance of the LSTM, GRU, and GTA models in predicting outcomes across different drive cycles (HWFET, LA92, UDDS, and US06) and temperature conditions (0 °C, 10 °C, and 25 °C). Among the three, the GTA model stands out, consistently achieving the lowest RMSE and MAE values in every scenario. This indicates that it is significantly more accurate and reliable than both LSTM and the GRU. Notably, all models perform better as the temperature increases, with GTA showing the most substantial improvement. For instance, in the HWFET cycle at 0 °C, GTA records an RMSE of just 1.08% compared with 4.63% for LSTM and 4.68% for the GRU. Similarly, in the LA92 cycle, GTA delivers an RMSE of 0.61% at 0 °C, outperforming LSTM and the GRU, which have errors exceeding 5%. This trend continues across the UDDS and US06 cycles, where GTA maintains its accuracy regardless of temperature. While LSTM generally performs slightly better than the GRU, both models fall short when compared with the GTA, especially in terms of consistency. Overall, the GTA model proves to be the most effective, making it the ideal choice for applications that require precise predictions across various conditions.

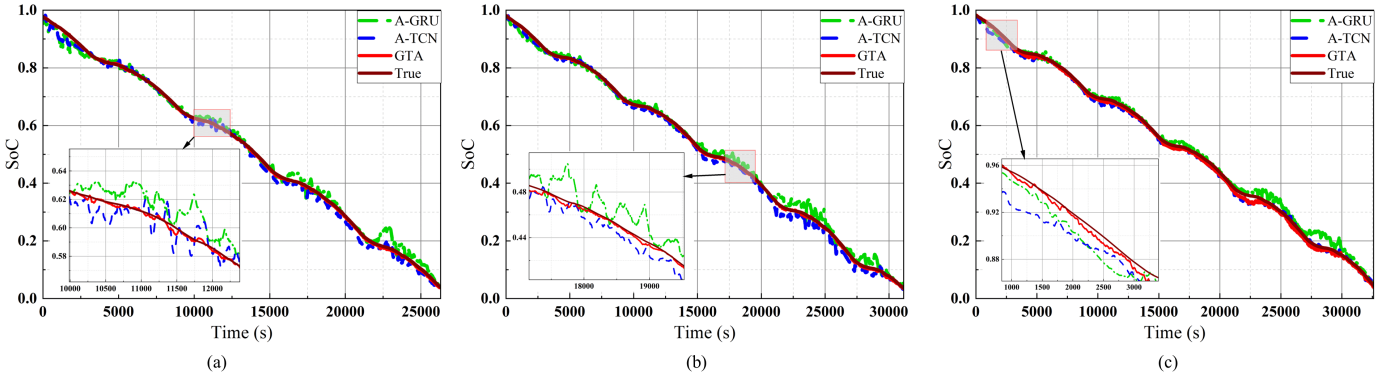


Figure 17. Model performance comparisons with A-TCN and A-GRU for drive cycle US06 at (**a**) 0 °C, (**b**) 10 °C, and (**c**) 25 °C.

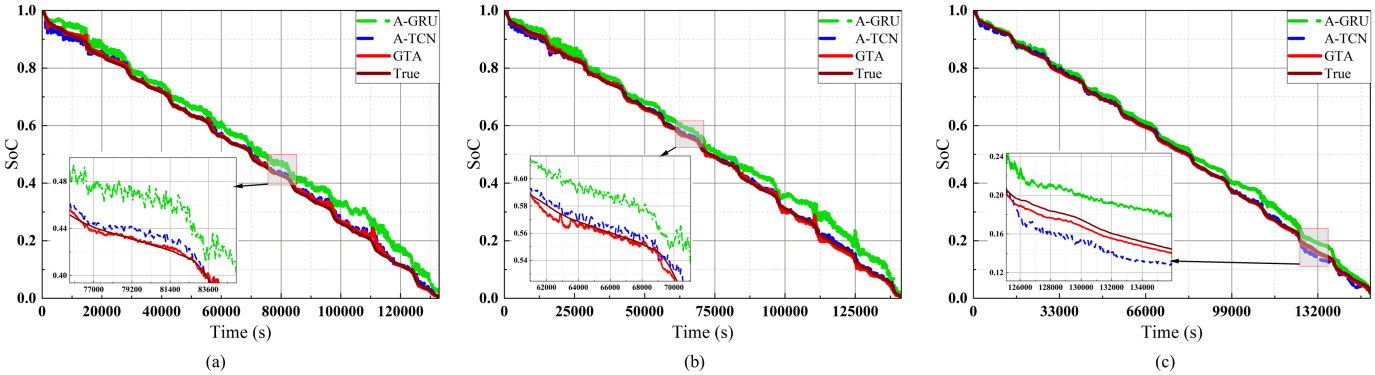


Figure 18. Model performance comparisons with A-TCN and A-GRU for drive cycle UDDS at (**a**) 0 °C, (**b**) 10 °C, and (**c**) 25 °C.

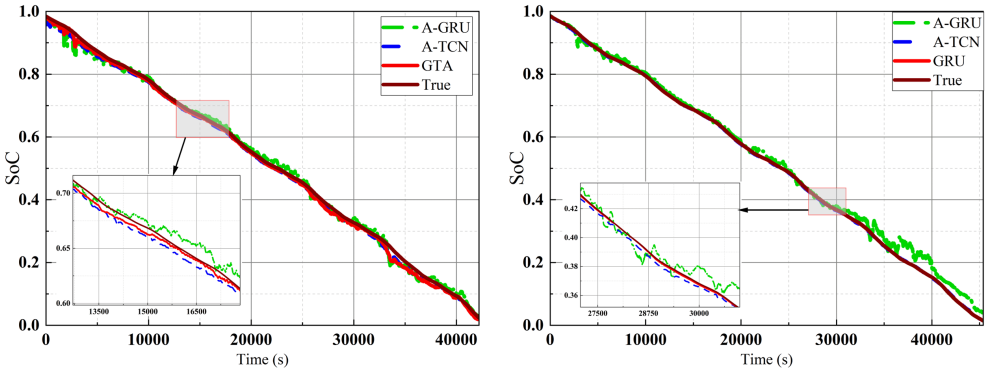


Figure 19. Model performance comparisons with A-TCN and A-GRU for drive cycle HWFET at (**a**) 0 °C and (**b**) 10 °C.

Table 6. Performance comparison between GTA and A-TCN.

Drive Cycle	Temperature (°C)	Model	RMSE (%)	MAE (%)	MaxE (%)
LA92	0	A-GRU	3.5	3.11	9.3
		A-TCN	0.96	0.68	5.9
		GTA	**0.612**	**0.418**	**4.7**
	10	A-GRU	2.9	2.14	9.81
		A-TCN	1.17	0.91	4.96
		GTA	**0.29**	**0.24**	**1.3**
	25	A-GRU	2.43	2.03	6.09
		A-TCN	0.73	0.57	2.54
		GTA	**0.45**	**0.36**	**1.75**
US06	0	A-GRU	1.82	1.3	7.23
		A-TCN	1.48	1.16	4.11
		GTA	**0.28**	**0.22**	**1.06**
	10	A-GRU	1.57	1.16	6.28
		A-TCN	1.46	1.2	5.37
		GTA	**0.26**	**0.21**	**0.9**
	25	A-GRU	1.59	1.14	5.71
		A-TCN	1.2	0.97	3.37
		GTA	**0.89**	**0.77**	**2.8**
UDDS	0	A-GRU	4.42	4.02	10.34
		A-TCN	1.17	0.89	5.61
		GTA	**0.74**	**0.47**	**5.54**
	10	A-GRU	3.6	3.16	9.95
		A-TCN	0.7	0.55	3.01
		GTA	**0.55**	**0.4**	**2.98**
	25	A-GRU	2.22	1.86	5.65
		A-TCN	0.84	0.6	2.8
		GTA	**0.33**	**0.27**	**0.8**
HWFET	0	A-GRU	1.42	1.05	7.19
		A-TCN	1.2	1.06	**3.07**
		GTA	**1.08**	**0.85**	4.25
	10	A-GRU	0.2	0.14	0.7
		A-TCN	0.21	0.18	0.6
		GTA	**0.16**	**0.11**	**0.4**

Table 7. Performance comparison with traditional models.

Drive Cycle	Temperature (°C)	Model	RMSE (%)	MAE (%)
HWFET	0	LSTM	4.63	4.00
		GRU	4.68	3.97
		GTA	1.08	0.85
	10	LSTM	4.93	4.23
		GRU	5.01	4.21
		GTA	0.16	0.11
LA92	0	LSTM	5.64	4.92
		GRU	5.72	5.08
		GTA	0.61	0.41
	10	LSTM	6.67	5.96
		GRU	6.01	5.81
		GTA	0.29	0.24
	25	LSTM	5.56	4.89
		GRU	5.75	5.06
		GTA	0.45	0.36
UDDS	0	LSTM	6.14	5.37
		GRU	5.41	4.49
		GTA	0.74	0.47
	10	LSTM	5.85	5.02
		GRU	6.55	5.65
		GTA	0.55	0.40
	25	LSTM	5.56	4.82
		GRU	5.86	4.99
		GTA	0.33	0.27
US06	0	LSTM	4.97	4.18
		GRU	5.41	4.21
		GTA	0.28	0.22
	10	LSTM	4.91	4.37
		GRU	4.87	4.28
		GTA	0.26	0.21
	25	LSTM	4.93	3.93
		GRU	5.01	4.11
		GTA	0.89	0.77

The model's performance is evaluated through a comparative analysis with recent studies in the field, as shown in Table 8, focusing on RMSE, MAE, and MaxE metrics. Results indicate a notable enhancement in model accuracy compared with previous research, reflecting improved performance across various evaluation criteria. Despite these limitations, it is worth noting that the proposed approach regularly beats the A-TCN across a variety of scenarios, demonstrating its dependability and effectiveness in SoC estimates. This highlights the model's design, which incorporates the features of the GRU and TCN architectures, providing improved capabilities for efficiently capturing sequences and increasing estimation accuracy.

The reason behind the better performance of the proposed GTA model can be attributed to several key factors that distinguish it from the A-TCN and A-GRU models and other recent studies. Firstly, the new approach of using the GRU and TCN in parallel with inter-layer attention improves the capturing capacity of complex, nonlinear patterns of SoC variations, which traditional models often struggle with. The A-TCN model can follow the long-term sequence but suffers in the short-term sequence as it exhibits oscillations, and A-GRU can capture the short-term sequence but shows a deviation in the long-term sequence. Therefore, when those two models are used in parallel, both models balance the

global trends and provide a more stable SoC estimation. On the other hand, with features extracted from both networks, the proposed models gain a richer representation of input data, which enhances SoC estimation accuracy. Additionally, the proposed model is generalized under different drive cycles and temperature conditions. Though there are some limitations in the lower temperature range, the GTA model provides a more accurate and stable SoC estimation, making it a potential solution for battery SoC estimation.

Table 8. Performance comparison with the latest study.

Methods	RMSE (%)	MAE (%)	MaxE (%)
LSTM encoder–decoder [50]	-	1.07	4.62
TCN-LSTM [51]	0.81	0.70	2.6
QTGA [54]	0.96	0.80	3.12
Proposed	**0.512**	**0.354**	**1.98**

4. Conclusions

This paper proposed a hybrid method with the help of the GRU, TCN, and attention mechanism to estimate battery SoC with high accuracy at different temperatures and in different drive cycles. With the ability to capture the long-term trend by the TCN and dynamically handle the short-term sequence by the GRU, the proposed model improves the model estimation accuracy and temporal flexibility, and reduces overfitting. The use of attention enhances the model's efficiency and moving averages aid immunity to the transient effect of voltage and current. Furthermore, the proposed model is tested under different drive cycles and temperature conditions. The performance of the GTA model is better than the traditional A-TCN and A-GRU models, where both RMSE and MAE are less than 1.08% and 0.85%, respectively, whereas MaxE is less than 5.54%. The model fitness score (R^2 score) is found to be more than 99.84%, which denotes that the model is more generalized as it can estimate a more stable SoC with high accuracy in different drive cycles and temperature ranges. In the case of the noisy input model, this also shows a stable SoC estimation capability, and overall estimation error is kept less than 2.5% in high noise conditions, showing the noise-resilient capability of the proposed model. On the other hand, the model performance is compared with A-TCN and A-GRU models and it is found that the model performance is better than other models. While the proposed model outperforms existing approaches, further investigation is needed for extreme temperature conditions (e.g., below 0 °C or above 40 °C). In these scenarios, rapid changes in aging effects and electrolyte viscosity significantly impact capacity and introduce measurement errors, affecting SoC estimation accuracy. Exploring these challenges could offer valuable insights into the model's adaptability and robustness under more demanding operating environments.

Author Contributions: Conceptualization, M.S.N.; Methodology, M.S.N.; Software, M.S.N.; Validation, M.S.N.; Formal analysis, M.S.N. and M.M.R.; Investigation, M.S.N.; Data curation, M.I.J.; Writing—original draft, M.S.N.; Writing—review & editing, M.M.R. and M.I.J.; Visualization, M.M.R.; Supervision, Y.M.J.; Project administration, Y.M.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea Government (MSIT) (No. 2022R1A2C1007884).

Data Availability Statement: The original data presented in the study are openly available in Mendeley data at https://data.mendeley.com/datasets/cp3473x7xv/3 (accessed on 7 November 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Li, F.; Zuo, W.; Zhou, K.; Li, Q.; Huang, Y. State of charge estimation of lithium-ion batteries based on PSO-TCN-Attention neural network. *J. Energy Storage* **2024**, *84*, 110806. [CrossRef]
2. Shrivastava, P.; Soon, T.K.; Idris, M.Y.I.B.; Mekhilef, S. Overview of model-based online state-of-charge estimation using Kalman filter family for lithium-ion batteries. *Renew. Sustain. Energy Rev.* **2019**, *113*, 109233. [CrossRef]
3. Shao, Y.; Zheng, Y.; Zhang, J.; Han, X.; Jin, B.; Sun, Y. A cloud capacity estimation method for electric vehicle lithium-ion battery independent of cloud SOC. *J. Energy Storage* **2024**, *85*, 110998. [CrossRef]
4. Huang, Q.; Li, J.; Xu, Q.; He, C.; Yang, C.; Cai, L.; Xu, Q.; Xiang, L.; Zou, X.; Li, X. State of Charge Estimation in Batteries for Electric Vehicle Based on Levenberg–Marquardt Algorithm and Kalman Filter. *World Electr. Veh. J.* **2024**, *15*, 391. [CrossRef]
5. Wu, L.; Lyu, Z.; Huang, Z.; Zhang, C.; Wei, C. Physics-based battery SOC estimation methods: Recent advances and future perspectives. *J. Energy Chem.* **2023**, *89*, 27–40. [CrossRef]
6. 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. [CrossRef]
7. Feng, Y.; Xue, C.; Han, F.; Cao, Z.; Yang, R.J. State-of-Charge and State-of-Health Estimation in Li-Ion Batteries Using Cascade Electrochemical Model-Based Sliding-Mode Observers. *Batteries* **2024**, *10*, 290. [CrossRef]
8. Zhao, J.; Hu, Z.; Wang, H.; Yu, K.; Zou, W.; Pan, T.; Mao, L. A multi-scale SOC estimation method for lithium-ion batteries incorporating expansion force. *J. Energy Storage* **2024**, *82*, 110481. [CrossRef]
9. Song, X.; Yang, F.; Wang, D.; Tsui, K.L. Combined CNN-LSTM network for state-of-charge estimation of lithium-ion batteries. *IEEE Access* **2019**, *7*, 88894–88902. [CrossRef]
10. Sesidhar, D.; Badachi, C.; Green II, R.C. A review on data-driven SOC estimation with Li-Ion batteries: Implementation methods & future aspirations. *J. Energy Storage* **2023**, *72*, 108420. [CrossRef]
11. Chang, Y.; Li, R.; Sun, H.; Zhang, X. Estimation of SOC in Lithium-Iron-Phosphate Batteries Using an Adaptive Sliding Mode Observer with Simplified Hysteresis Model during Electric Vehicle Duty Cycles. *Batteries* **2024**, *10*, 154. [CrossRef]
12. Qian, C.; Guan, H.; Xu, B.; Xia, Q.; Sun, B.; Ren, Y.; Wang, Z. A CNN-SAM-LSTM hybrid neural network for multi-state estimation of lithium-ion batteries under dynamical operating conditions. *Energy* **2024**, *294* , 130764. [CrossRef]
13. Wang, Y.; Tian, J.; Sun, Z.; Wang, L.; Xu, R.; Li, M.; Chen, Z. A comprehensive review of battery modeling and state estimation approaches for advanced battery management systems. *Renew. Sustain. Energy Rev.* **2020**, *131*, 110015. [CrossRef]
14. Hassan, M.U.; Saha, S.; Haque, M.E.; Islam, S.; Mahmud, A.; Mendis, N. A comprehensive review of battery state of charge estimation techniques. *Sustain. Energy Technol. Assess.* **2022**, *54*, 102801. [CrossRef]
15. Hossain, M.; Haque, M.; Arif, M.T. Kalman filtering techniques for the online model parameters and state of charge estimation of the Li-ion batteries: A comparative analysis. *J. Energy Storage* **2022**, *51*, 104174. [CrossRef]
16. 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 Mater.* **2023**, *61*, 102883. [CrossRef]
17. Zheng, Y.; Ouyang, M.; Han, X.; Lu, L.; Li, J. Investigating the error sources of the online state of charge estimation methods for lithium-ion batteries in electric vehicles. *J. Power Sources* **2018**, *377*, 161–188. [CrossRef]
18. Lin, C.; Tang, A.; Xing, J. Evaluation of electrochemical models based battery state-of-charge estimation approaches for electric vehicles. *Appl. Energy* **2017**, *207*, 394–404. [CrossRef]
19. Lai, X.; Zheng, Y.; Sun, T. A comparative study of different equivalent circuit models for estimating state-of-charge of lithium-ion batteries. *Electrochim. Acta* **2018**, *259*, 566–577. [CrossRef]
20. Corno, M.; Bhatt, N.; Savaresi, S.M.; Verhaegen, M. Electrochemical model-based state of charge estimation for Li-ion cells. *IEEE Trans. Control Syst. Technol.* **2014**, *23*, 117–127. [CrossRef]
21. Zhou, W.; Zheng, Y.; Pan, Z.; Lu, Q. Review on the battery model and SOC estimation method. *Processes* **2021**, *9*, 1685. [CrossRef]
22. He, H.; Xiong, R.; Fan, J. Evaluation of lithium-ion battery equivalent circuit models for state of charge estimation by an experimental approach. *Energies* **2011**, *4*, 582–598. [CrossRef]
23. Xiong, R.; Tian, J.; Shen, W.; Sun, F. A novel fractional order model for state of charge estimation in lithium ion batteries. *IEEE Trans. Veh. Technol.* **2018**, *68*, 4130–4139. [CrossRef]
24. Tian, J.; Xiong, R.; Shen, W.; Wang, J.; Yang, R. Online simultaneous identification of parameters and order of a fractional order battery model. *J. Clean. Prod.* **2020**, *247*, 119147. [CrossRef]
25. How, D.N.; Hannan, M.; Lipu, M.H.; Ker, P.J. State of charge estimation for lithium-ion batteries using model-based and data-driven methods: A review. *IEEE Access* **2019**, *7*, 136116–136136. [CrossRef]
26. Martí-Florences, M.; Cecilia, A.; Clemente, A.; Costa-Castelló, R. SoC Estimation in Lithium-Ion Batteries with Noisy Measurements and Absence of Excitation. *Batteries* **2023**, *9*, 578. [CrossRef]
27. Wang, W.; Mu, J. State of charge estimation for lithium-ion battery in electric vehicle based on Kalman filter considering model error. *IEEE Access* **2019**, *7*, 29223–29235. [CrossRef]
28. Lee, J.; Nam, O.; Cho, B. Li-ion battery SOC estimation method based on the reduced order extended Kalman filtering. *J. Power Sources* **2007**, *174*, 9–15. [CrossRef]
29. Guo, Y.; Tian, J.; Li, X.; Song, B.; Tian, Y. State of charge estimation of lithium-ion batteries based on vector forgetting factor recursive least square and improved adaptive cubature kalman filter. *Batteries* **2023**, *9*, 499. [CrossRef]

30. Peng, J.; Luo, J.; He, H.; Lu, B. An improved state of charge estimation method based on cubature Kalman filter for lithium-ion batteries. *Appl. Energy* **2019**, *253*, 113520. [CrossRef]
31. Tulsyan, A.; Tsai, Y.; Gopaluni, R.B.; Braatz, R.D. State-of-charge estimation in lithium-ion batteries: A particle filter approach. *J. Power Sources* **2016**, *331*, 208–223. [CrossRef]
32. Li, J.; Barillas, J.K.; Guenther, C.; Danzer, M.A. A comparative study of state of charge estimation algorithms for LiFePO4 batteries used in electric vehicles. *J. Power Sources* **2013**, *230*, 244–250. [CrossRef]
33. Lipu, M.H.; Hannan, M.; Hussain, A.; Ayob, A.; Saad, M.H.; Karim, T.F.; How, D.N. Data-driven state of charge estimation of lithium-ion batteries: Algorithms, implementation factors, limitations and future trends. *J. Clean. Prod.* **2020**, *277*, 124110. [CrossRef]
34. Oji, T.; Zhou, Y.; Ci, S.; Kang, F.; Chen, X.; Liu, X. Data-driven methods for battery soh estimation: Survey and a critical analysis. *IEEE Access* **2021**, *9*, 126903–126916. [CrossRef]
35. Lai, X.; Yi, W.; Cui, Y.; Qin, C.; Han, X.; Sun, T.; Zhou, L.; Zheng, Y. Capacity estimation of lithium-ion cells by combining model-based and data-driven methods based on a sequential extended Kalman filter. *Energy* **2021**, *216*, 119233. [CrossRef]
36. Khaleghi, S.; Hosen, M.S.; Karimi, D.; Behi, H.; Beheshti, S.H.; Van Mierlo, J.; Berecibar, M. Developing an online data-driven approach for prognostics and health management of lithium-ion batteries. *Appl. Energy* **2022**, *308*, 118348. [CrossRef]
37. Cai, L.; Meng, J.; Stroe, D.I.; Peng, J.; Luo, G.; Teodorescu, R. Multiobjective optimization of data-driven model for lithium-ion battery SOH estimation with short-term feature. *IEEE Trans. Power Electron.* **2020**, *35*, 11855–11864. [CrossRef]
38. Sahinoglu, G.O.; Pajovic, M.; Sahinoglu, Z.; Wang, Y.; Orlik, P.V.; Wada, T. Battery state-of-charge estimation based on regular/recurrent Gaussian process regression. *IEEE Trans. Ind. Electron.* **2017**, *65*, 4311–4321. [CrossRef]
39. Hu, J.; Hu, J.; Lin, H.; Li, X.; Jiang, C.; Qiu, X.; Li, W. State-of-charge estimation for battery management system using optimized support vector machine for regression. *J. Power Sources* **2014**, *269*, 682–693. [CrossRef]
40. Anton, J.C.A.; Nieto, P.J.G.; Viejo, C.B.; Vilán, J.A.V. Support vector machines used to estimate the battery state of charge. *IEEE Trans. Power Electron.* **2013**, *28*, 5919–5926. [CrossRef]
41. Klass, V.; Behm, M.; Lindbergh, G. A support vector machine-based state-of-health estimation method for lithium-ion batteries under electric vehicle operation. *J. Power Sources* **2014**, *270*, 262–272. [CrossRef]
42. Fan, X.; Zhang, W.; Zhang, C.; Chen, A.; An, F. SOC estimation of Li-ion battery using convolutional neural network with U-Net architecture. *Energy* **2022**, *256*, 124612. [CrossRef]
43. Bhattacharjee, A.; Verma, A.; Mishra, S.; Saha, T.K. Estimating state of charge for xEV batteries using 1D convolutional neural networks and transfer learning. *IEEE Trans. Veh. Technol.* **2021**, *70*, 3123–3135. [CrossRef]
44. Zhang, D.; Zhong, C.; Xu, P.; Tian, Y. Deep learning in the state of charge estimation for li-ion batteries of electric vehicles: A review. *Machines* **2022**, *10*, 912. [CrossRef]
45. Hannan, M.A.; How, D.N.; Lipu, M.H.; Ker, P.J.; Dong, Z.Y.; Mansur, M.; Blaabjerg, F. SOC estimation of li-ion batteries with learning rate-optimized deep fully convolutional network. *IEEE Trans. Power Electron.* **2020**, *36*, 7349–7353. [CrossRef]
46. Cui, Z.; Wang, L.; Li, Q.; Wang, K. A comprehensive review on the state of charge estimation for lithium-ion battery based on neural network. *Int. J. Energy Res.* **2022**, *46*, 5423–5440. [CrossRef]
47. Hong, J.; Wang, Z.; Chen, W.; Wang, L.Y.; Qu, C. Online joint-prediction of multi-forward-step battery SOC using LSTM neural networks and multiple linear regression for real-world electric vehicles. *J. Energy Storage* **2020**, *30*, 101459. [CrossRef]
48. Zhao, F.; Guo, Y.; Chen, B. A Review of Lithium-Ion Battery State of Charge Estimation Methods Based on Machine Learning. *World Electr. Veh. J.* **2024**, *15*, 131. [CrossRef]
49. Xiao, B.; Liu, Y.; Xiao, B. Accurate state-of-charge estimation approach for lithium-ion batteries by gated recurrent unit with ensemble optimizer. *IEEE Access* **2019**, *7*, 54192–54202. [CrossRef]
50. Bian, C.; He, H.; Yang, S.; Huang, T. State-of-charge sequence estimation of lithium-ion battery based on bidirectional long short-term memory encoder-decoder architecture. *J. Power Sources* **2020**, *449*, 227558. [CrossRef]
51. Hu, C.; Cheng, F.; Ma, L.; Li, B. State of charge estimation for lithium-ion batteries based on TCN-LSTM neural networks. *J. Electrochem. Soc.* **2022**, *169*, 030544. [CrossRef]
52. Guo, S.; Ma, L. A comparative study of different deep learning algorithms for lithium-ion batteries on state-of-charge estimation. *Energy* **2023**, *263*, 125872. [CrossRef]
53. Huang, Z.; Yang, F.; Xu, F.; Song, X.; Tsui, K.L. Convolutional gated recurrent unit–recurrent neural network for state-of-charge estimation of lithium-ion batteries. *IEEE Access* **2019**, *7*, 93139–93149. [CrossRef]
54. Li, H.; Fu, L.; Long, X.; Liu, L.; Zeng, Z. A hybrid deep learning model for lithium-ion batteries state of charge estimation based on quantile regression and attention. *Energy* **2024**, *294* , 130834. [CrossRef]
55. Wei, Z.; Zhao, D.; He, H.; Cao, W.; Dong, G. A noise-tolerant model parameterization method for lithium-ion battery management system. *Appl. Energy* **2020**, *268*, 114932. [CrossRef]
56. Lin, X. Theoretical analysis of battery SOC estimation errors under sensor bias and variance. *IEEE Trans. Ind. Electron.* **2018**, *65*, 7138–7148. [CrossRef]
57. Kollmeyer, P.; Vidal, C.; Naguib, M.; Skells, M. *LG 18650HG2 Li-ion Battery Data*; Kaggle: San Francisco, CA, USA, 2023. [CrossRef]
58. Vidal, C.; Kollmeyer, P.; Naguib, M.; Malysz, P.; Gross, O.; Emadi, A. Robust xev battery state-of-charge estimator design using a feedforward deep neural network. *SAE Int. J. Adv. Curr. Pract. Mobil.* **2020**, *2*, 2872–2880. [CrossRef]

59. Lea, C.; Vidal, R.; Reiter, A.; Hager, G.D. Temporal convolutional networks: A unified approach to action segmentation. In Proceedings of the Computer Vision–ECCV 2016 Workshops, Amsterdam, The Netherlands, 8–10 October and 15–16 October 2016; pp. 47–54. [CrossRef]
60. Bai, S.; Kolter, J.Z.; Koltun, V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv* **2018**, arXiv:1803.01271. [CrossRef]
61. Lara-Benítez, P.; Carranza-García, M.; Luna-Romera, J.M.; Riquelme, J.C. Temporal convolutional networks applied to energy-related time series forecasting. *Appl. Sci.* **2020**, *10*, 2322. [CrossRef]
62. Yan, J.; Mu, L.; Wang, L.; Ranjan, R.; Zomaya, A.Y. Temporal convolutional networks for the advance prediction of ENSO. *Sci. Rep.* **2020**, *10*, 8055. [CrossRef]
63. Yang, X.; Hu, J.; Hu, G.; Guo, X. Battery state of charge estimation using temporal convolutional network based on electric vehicles operating data. *J. Energy Storage* **2022**, *55*, 105820. [CrossRef]
64. Chen, J.; Zhang, Y.; Li, W.; Cheng, W.; Zhu, Q. State of charge estimation for lithium-ion batteries using gated recurrent unit recurrent neural network and adaptive Kalman filter. *J. Energy Storage* **2022**, *55*, 105396. [CrossRef]
65. Wang, Y.X.; Chen, Z.; Zhang, W. Lithium-ion battery state-of-charge estimation for small target sample sets using the improved GRU-based transfer learning. *Energy* **2022**, *244*, 123178. [CrossRef]
66. Tian, Y.; Lai, R.; Li, X.; Tian, J. State-of-charge estimation for lithium-ion batteries based on attentional sequence-to-sequence architecture. *J. Energy Storage* **2023**, *62*, 106836. [CrossRef]
67. Tang, A.; Jiang, Y.; Yu, Q.; Zhang, Z. A hybrid neural network model with attention mechanism for state of health estimation of lithium-ion batteries. *J. Energy Storage* **2023**, *68*, 107734. [CrossRef]
68. Li, H.; Min, M.R.; Ge, Y.; Kadav, A. A Context-aware Attention Network for Interactive Question Answering. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '17, New York, NY, USA, 13–17 August 2017; pp. 927–935. [CrossRef]
69. Vapnik, V. *The Nature of Statistical Learning Theory*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013.
70. Meyer, G.P. An alternative probabilistic interpretation of the huber loss. In Proceedings of the IEEE/CVF Conference on Computer Vision And Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 5261–5269. [CrossRef]