Chapter 3

Interpretable Deep Learning Framework for Battery SOC Prediction

Lithium-ion batteries are the driving force behind electric vehicles and portable electronic devices. Accurate estimation of the state of charge in lithium-ion batteries is crucial for optimizing battery performance and improving energy efficiency. This chapter proposes a novel hybrid model that combines a multi-head dilated temporal convolutional network architecture with a gated recurrent unit to anticipate the state of charge levels. The novel multi-head architecture of the dilated temporal convolutional network facilitates simultaneous learning of patterns across different scales, allowing the model to adapt to new patterns quickly. The diverse dilation rates in the dilated temporal convolutional network enhance the model's capability to capture long-term sequences, while the gated recurrent unit focuses on short-term dependencies, offering a versatile state of charge estimation method suitable for various environmental conditions. Additionally, the incorporation of the explainable artificial intelligence technique - Shapley Additive exPlanations aids in achieving global interpretability for state of charge prediction, offering

a precise quantification of the influence of individual attributes. Comprehensive experiments were conducted across various temperatures and driving cycles to demonstrate the effectiveness of the proposed model. The computation results indicate the proposed method's adaptability to varying conditions, achieving high estimation accuracy and robustness with a mean absolute percentage error and root mean square percentage error of 0.54% and 0.84%, respectively, along with a parameter count of 3,74,433. Moreover, the proposed architecture enhances state of charge estimation performance compared to existing models across multiple datasets while maintaining a more efficient parameter count.

3.1 Introduction

Accurate state of charge (SOC) estimation is crucial for optimizing the performance of lithium-ion batteries (LIBs) in electric vehicles (EVs), ensuring efficient energy use while minimizing environmental impact. By precisely monitoring SOC, battery life can be extended, reducing the need for frequent replacements and lowering the ecological footprint associated with the raw material extraction. This not only enhances EV efficiency but also supports broader environmental goals by cutting greenhouse gas emissions and protecting ecosystems, paving the way for a cleaner, greener transportation future.

The commonly used SOC estimation techniques consist of look-up tables [98], [99], model-based approaches [100],[101], and coulomb counting method [102], [103]. As direct measurement of SOC is complicated due to its non-linear and time-varying behaviour, data-driven methods are gaining popularity among researchers [104],[105]. Look-up table-based techniques primarily involve utilizing parameters such as open circuit voltage and impedance. Estimating the SOC by referencing a table that encompasses these battery characteristics is possible, but collecting these parameters in real-time is a considerable challenge. A potential drawback of the mentioned approach is its dependency on pre-

established look-up tables, which might not accurately capture complex and dynamic battery behaviours. A model-based approach for SOC estimation [106] uses either or both of the battery's electrical and chemical properties and models it accordingly. These models are represented as state equations. The major drawback of this approach is the need for domain knowledge and the time required to develop these models. One of the most widely employed techniques for estimating SOC is the coulomb counting method. This approach involves integrating the discharge currents over time. However, its open-loop nature can lead to inherent errors, especially if the initial SOC is imprecise and the current sensors are inaccurate.

Data-driven methods are black-box techniques focusing on the relationship between input variables and resulting outcomes in SOC estimation within a BMS. These methods do not make specific assumptions about the battery's inner workings. With the continuous improvement of hardware capabilities, data-driven methods have made substantial strides in SOC estimation, and utilizing these methods eliminates the necessity for an extensive theoretical understanding of LIBs. Artificial intelligence based methods are entirely data-driven and use machine learning [107],[108] algorithms or deep learning systems [109], [110], [111] and also transfer learning techniques [112],[113],[114] to estimate the SOC effectively. In contrast to typical machine learning methods, deep learning technologies employ various techniques to extract a higher level of information from the raw input [115]. These systems can handle large volumes of data by using a deeper network. This capability of deep learning systems reduces the necessity of manual feature engineering and minimizes the reliance on labeled data.

The SOC relies not solely on current measurements but is also strongly associated with historical data. Deep Learning systems involving recurrent neural networks (RNN) and their extension can effectively leverage past information and present data to predict the current output values [116]. The major drawback of traditional RNN is the vanishing gradient problem in long-term sequences. Gated architectures of long short-term memory

(LSTM) and gated recurrent units (GRUs) can be used to address these issues. GRUs are effective in capturing and propagating information over extended temporal sequences. With fewer parameters, simplified architecture, faster training, and enhanced memory efficiency compared to LSTM networks, GRUs emerge as a compelling choice for accurate SOC predictions [117].

Convolutional neural networks (CNNs), a type of deep learning model, are used for SOC estimation because they can learn hierarchical features from spatial data [118]. However, a drawback of CNNs lies in their limited capability to capture long-range dependencies in sequential data [119]. Temporal convolutional networks (TCNs) offer efficient parallel processing for battery data and demonstrate superior performance over CNNs in capturing long-range dependencies in temporal sequences. However, they may still have limitations in capturing complex patterns and trends over extended periods[120]. The proposed multi-head dilated temporal convolutional network (MHDTCN) addresses this limitation by leveraging a novel architecture that excels at recognizing intricate patterns and trends over extended periods by using varying dilation rates. By incorporating multiple branches with dilated TCN blocks, the MHDTCN architecture can capture diverse data features and achieve more comprehensive temporal modeling. This leads to improved SOC estimation accuracy.

Existing methods lack a comprehensive approach to temporal modeling, focusing either on short-term dynamics or long-term trends in battery data. This limitation can lead to inaccurate SOC estimations, which are critical for the efficient operation and management of battery systems, particularly in EVs. The proposed hybrid model aims to overcome this limitation by integrating the strengths of two distinct architectures: the novel MHDTCN and the GRU. The MHDTCN is proficient in recognizing intricate long-term patterns and trends in data, while the GRU excels at capturing short-term dependencies. By synergizing these architectures, the hybrid model achieves a holistic approach to temporal modeling. This enables it to accurately capture nuanced short-

term dynamics as well as broader long-term trends in battery data. Hybrid methods are popular for SOC estimation because they improve performance by combining the strengths of various machine learning algorithms and deep learning techniques. [121],[122].

Furthermore, the utilization of a multi-head architecture in the proposed technique further enhances its capabilities. By implementing several branches, each featuring its array of dilated TCN blocks, the model can effectively grasp various levels of detail within the data, spanning multiple scales. The outputs from these branches are concatenated, allowing the model to leverage complementary information from different heads. The GRU, operating in conjunction with the MHDTCN, is pivotal in capturing intricate sequence patterns, contributing to the model's overall efficacy. Incorporating explainable artificial intelligence (XAI) techniques such as SHapley Additive exPlanations (SHAP) in the proposed study aims to enhance the trustworthiness and understanding of the SOC estimation model, particularly for practical implementation in real-world EV applications. Ensuring interpretability is crucial, as it enables stakeholders to grasp the rationale behind the SOC estimate model's conclusions. Therefore, openness and understandability in SOC estimation models are essential for fostering confidence among stakeholders and consumers, mitigating risks, and ensuring battery-powered systems' dependable and secure operation. The major contributions of the proposed work in this chapter are:

- Introducing a novel MHDTCN architecture offers the primary benefit of capturing multiscale features within the data. Each head, with its specific dilation rate, focuses on different temporal resolutions, allowing the model to simultaneously learn patterns at various scales.
- The MHDTCN with GRU combines the strengths of both convolutional and recurrent architectures, thereby capturing both long-term and short-term temporal dependencies. This enables the model to capture complex relationships under

diverse environmental conditions and varied driving cycles for EVs, leading to improved mean absolute error (MAE) and root mean square error (RMSE) values compared to state-of-the-art methods.

- The proposed hybrid model excels in SOC estimation with a streamlined architecture with fewer parameters. By employing efficient parallel processing and diverse dilation rates through a novel MHDTCN architecture, the model achieves accelerated learning, significantly improving computational efficiency and reducing resource requirements for real-world applications.
- Integrating explainable XAI techniques like SHAP for SOC estimation in EVs enhances model transparency and interpretability. This integration facilitates the identification of crucial input features under varying ambient temperatures.

3.2 Proposed Methodology

TCN is an extension of CNN architecture specially designed for sequence modeling tasks. It works on the principle of convolution itself and uses causal convolution and zero padding. The major drawback of traditional TCN is the vanishing gradient problem and its inability to handle large data sequences. To overcome these issues, dilated TCN is used, which uses dilated convolutions. Dilated convolutions have holes in their kernels, ensuring that each input within the history is covered by a filter, allowing the deep network to handle an extensive receptive history. The primary advantages of dilated TCNs are their increased receptive field, which is well-suited for capturing long-term dependencies, and their reduced computational cost due to fewer parameters. The output at time t in a dilated TCN convolves only with elements from time t and earlier of the previous layer. To capture temporal dependencies in a scalable and effective way, a dilated TCN unit combines dilated causal convolutions, rectified linear unit (ReLU) activation, normalization, and depth-wise convolution within a single 1-D convolution unit.

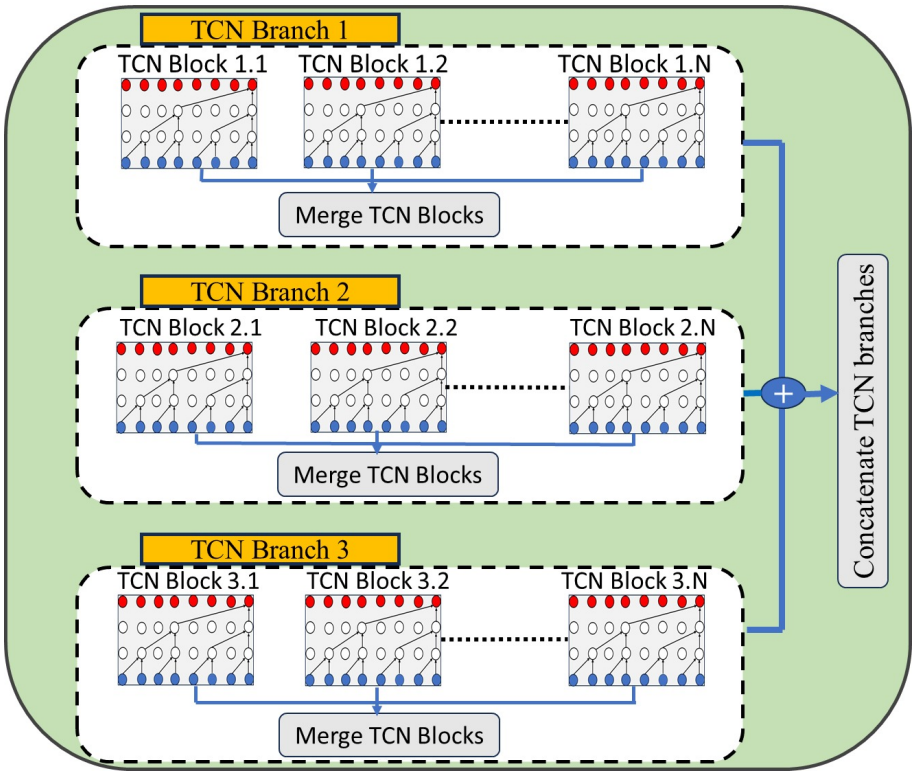


Figure 3.1: The architecture of MHDTCN.

3.2.1 Multi-Head Dilated Temporal Convolutional Network

The proposed MHDTCN features multiple branches that process data in parallel, as shown in Fig. 3.1. Each branch contains TCN blocks with different dilation rates. These dilated convolutions increase the model's receptive field, allowing it to capture information across various time scales. The varying dilation rates in different branches enable the model to learn both the short-term and long-term sequences. The TCN blocks from each branch are merged, allowing the model to integrate information learned at different temporal resolutions. Finally, the outputs of the multiple TCN branches are concatenated, thereby combining the information learned from different branches and varying dilation rates.

The foundation of MHDTCN architecture is the dilated TCN blocks, which consist of a series of dilated casual convolution layers followed by non-linear activations. Each

convolution layer captures information at varying temporal scales due to different dilation rates. The input sequence data is represented by X, which is a time series with T time steps, and X_t represents data at time step t. A dilated TCN block with a head h consists of N_h dilated convolution layers. Each convolution layer uses a different dilation rate $d_{h,n}$, where n is the layer index within the head. The output of each convolution layer is passed through a ReLU activation function. The mathematical representation of a single dilated block is as follows:

$$Y_{h,n,t} = \text{ReLU}\left(\sum_{i=1}^{k} W_{h,n,i} \cdot X_{t-d_{h,n,i}} + b_{h,n}\right) \tag{3.1}$$

where $Y_{h,n,t}$ represents the output of n^{th} convolution layer in head h, the kernel size is represented by k, $W_{h,n,i}$ denotes the weight of the convolution kernel at position i for layer n in head h, $X_{t-d_{h,n,i}}$ represent input data at time step t and $b_{h,n}$ represents the bias term for layer n in head h.

Multiple TCN branches are employed in the architecture, each with its own dilated TCN blocks. Within each head, the outputs of the N_h dilated convolutional layers are merged using element-wise addition:

$$Z_{h,t} = \sum_{n=1}^{N_h} Y_{h,n,t} \tag{3.2}$$

where $Z_{h,t}$ represents the output of head h at time step t.

The outputs of all H heads are further merged using element-wise addition to create a fused representation of the input data:

$$Z_t = \sum_{h=1}^{H} Z_{h,t} \tag{3.3}$$

where Z_t represents the fused output at time step t from all heads. The steps in our MHDTCN architecture are explained in Algorithm 1. So combining the above three

equations, the proposed multi-head TCN equation is as follows:

$$Z_t = \sum_{h=1}^{H} \sum_{n=1}^{N_h} \text{ReLU}\left(\sum_{i=1}^{k} W_{h,n,i} \cdot X_{t-d_{h,n,i}} + b_{h,n} \right) \tag{3.4}$$

Algorithm 1 Proposed MHDTCN Algorithm

Input: Sequential data $X = \{x_{itk}\}$, where i is the sample index, t represents the time steps, and k denotes the feature dimensions.

Initialization: Define parameters: H: Number of heads N_h: Number of dilated convolution layers in each head $D_{h,n}$: Dilation rates for each layer in each head K: Kernel size

MHDTCN Operations:

for $t = 1$ to T **do**
  for $h = 1$ to H **do**
    Initialize $Z_{h,t} = 0$
    /**Initialize head output for time step t.*
    for $n = 1$ to N_h **do**
      Compute the output of the dilated convolution layer:
      $Y_{h,n,t} = \text{ReLU}\left(\sum_{i=1}^{K} W_{h,n,i} \cdot X_{t-D_{h,n,i}} + b_{h,n}\right)$ /**Compute output of the n^{th} dilated convolution layer in head h at time step t.*
      Update the head output for time step t:
      $Z_{h,t} = Z_{h,t} + Y_{h,n,t}$
      /**Accumulate the output of the n^{th} layer in head h at time step t.*
    end for
    Finalize the fused output for time step t by combining outputs from all heads:
    $Z_t = Z_t + Z_{h,t}$
    /**Combine the outputs from all heads at time step t.*
  end for
end for

Output: Fused representation Z_t capturing diverse temporal patterns across all heads for each time step.

3.2.2 Gated Recurrent Unit

GRU is a variant of RNN architecture with specialized gating mechanisms. GRU is employed to capture additional temporal dependencies and facilitate sequential modeling. Two gating mechanisms used in the GRU are the update and reset gates. The update

gate determines how much of the previous state information will be used in the current state. The reset gate helps find the relevant previous hidden state information for the current time step calculation. The GRU is applied to the fused output Z_t to capture additional temporal dependencies. Let h_t represent the hidden state of the GRU layer at time step t. The following equations can describe the relationship between Z_t and h_t in the GRU layer:

$$res_t = \sigma(W_r \cdot Z_t + P_r \cdot h_{t-1} + bi_r) \tag{3.5}$$

$$up_t = \sigma(W_z \cdot Z_t + P_z \cdot h_{t-1} + bi_z) \tag{3.6}$$

$$\tilde{h}_t = \tanh(W_h \cdot Z_t + P_h \cdot (res_t \cdot h_{t-1}) + bi_h) \tag{3.7}$$

$$h_t = (1 - up_t) \cdot h_{t-1} + up_t \cdot \tilde{h}_t \tag{3.8}$$

where res_t, up_t, $\tilde{h}_t$, h_t represents reset gate, update gate, candidate hidden state, and new hidden state respectively. W_r, W_z, W_h are the weight matrices for the input Z_t; P_r, P_z, P_h are weight matrices for previous hidden state h_{t-1}; bi_r, bi_z, bi_h indicates the bias terms used. This set of equations illustrates how the multi-head architecture Z_t output is utilized as input to the GRU layer to compute hidden state h_t. The final output of the MHDTCN architecture obtained is used to find the hidden state of the GRU layer by applying a linear transformation.

3.2.3 Hybrid Model for SOC estimation

The proposed architecture combines dilated convolutions, multi-head merging, and GRU modeling to capture intricate patterns in sequential data. The overall architecture of the proposed system is shown in Fig. 3.2. The SOC is calculated by multiplying the current capacity by the maximum discharge capacity, producing a percentage representation of the battery's charge status. The dataset's essential input features for SOC estimate are voltage, current, capacitance, and temperature. These characteristics impact the

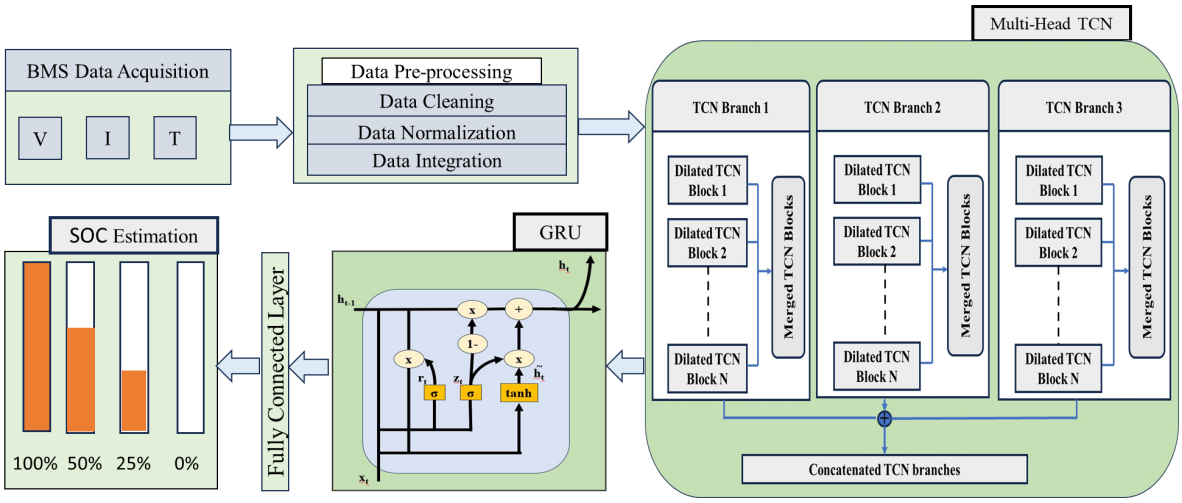


Figure 3.2: The framework of the proposed MHDTCN-GRU model.

battery's behaviour and the charge-discharge cycle.

Ensuring high data quality is fundamental in any machine learning application, starting with the data preprocessing phase. This phase involves identifying and eliminating missing values and outliers to ensure the model is trained on accurate and complete data, enhancing its robustness and reliability. The input features are normalized using min-max normalization to a consistent range, which facilitates efficient training. Bayesian Optimization is employed to fine-tune the hyperparameters of the hybrid model. This approach systematically explored key parameters such as the number of filters, kernel size, learning rate, dilation rate in MHDTCN, and number of units in GRU, aiming to maximize model performance measured by RMSE and MAE.

The proposed MHDTCN architecture employs three heads, each incorporating thirty-two filters. These filters operate with a kernel size of three, facilitating feature extraction from the input temporal sequence. Different dilation rates are applied within each head, allowing the proposed system to capture temporal dependencies at varying ranges. After the MHDTCN, the architecture incorporates a GRU layer with 256 units that further refine the features by capturing dynamic temporal relationships. By implementing a window size of 100 and a step size of 5, the proposed model adeptly captures long-term

patterns while ensuring a fine-grained analysis of the data. This approach balances the capture of detailed temporal dynamics with computational efficiency, allowing for more accurate detection of subtle trends and changes over time. The network concludes with a series of dense layers; the first dense layer contains 256 units, followed by a second dense layer with 128 units, culminating in an output dense layer designed for SOC estimation. The learning rate was optimized using a Bayesian optimization algorithm to achieve the best convergence and minimize loss, while the batch size was set to 32 to ensure model stability and computational efficiency.

In the MHDTCN architecture, convolutional layers are designed to recognize long-term patterns, such as voltage fluctuations, which aid in estimating the SOC by identifying behaviors indicative of specific battery states. The GRU layers play a crucial role in capturing short-term dependencies, which is particularly valuable in battery data analysis, where rapid voltage, current, or temperature fluctuations can signal critical events such as charging or discharging. Together, these layers complement each other, enabling the extraction of critical features essential for comprehensive battery data analysis. The proposed MHDTCN-GRU architecture excels in handling large time series datasets, effectively capturing long-term dependencies and subtle patterns in battery data. The MHDTCN-GRU architecture's primary challenge is its handling of low-dimensional datasets, as it finds it difficult to extract meaningful patterns and relationships due to limited available data. Additionally, while the architecture excels in temporal pattern extraction, it lacks tailored mechanisms for spatial feature extraction. There is potential for improvement by incorporating convolutional layers that operate in multi-dimensional mode. This enhancement would enable the model to capture spatial and temporal dependencies effectively.

The Battery Pack SOC estimation using Individual cell SOC method aligns well with the capabilities of the proposed hybrid architecture. The multi-head architecture of the MHDTCN enables simultaneous learning across different scales, crucial for capturing

variations and dynamics among multiple cells within the battery pack. incorporating diverse dilation rates, the algorithm effectively captures long-term sequences, such as those arising from factors like cell aging or gradual degradation, significantly influencing the overall pack SoC. Additionally, by providing individual cell SOC estimates, the proposed algorithm facilitates effective management of pack imbalances during charging and discharging cycles, enhancing the overall accuracy and reliability of pack-level SoC estimation

3.3 Results and Discussion

3.3.1 Experimental setup

The primary dataset utilized in the proposed work is the LG 18650HG2 Li-ion Battery Data sourced from the Mendeley data website [123], which provides a comprehensive view of battery performance under varying conditions. The battery data collection process involved meticulous tests spanning temperatures from -20°C to 40°C. Initially, the battery underwent a full charge using a constant current-constant voltage method at 1.5 A until reaching 4.2 V. Subsequent tests included four hybrid pulse power characterization tests covering discharge rates of 1C, 2C, 4C, and 6C, and charging rates of 0.5C, 1C, 1.5C, and 2C across SOC from 100% to 0%. Additional discharge tests were conducted at 0.5C, 2C, and two 1C rates before specific drive cycles. These cycles included the LA92 dynamometer driving schedule (LA92), US06 supplemental FTP driving schedule (US06), urban dynamometer driving schedule (UDDS), and highway fuel economy test (HWFET), along with eight mixed cycles (Mix 1–8) combining these profiles. The entire procedure was systematically repeated, reversing the temperature order: 25°C, 10°C, 0°C, and -10°C.

Another dataset used for SOC estimation and assessing the model's generalizability is the 72 real-world driving trips of a BMW i3 from the IEEE Dataport [124]. Of these

trips, 36 were recorded during the summer and the other 36 during the winter. This dataset includes forty-seven features related to weather, battery, vehicle, and heating and air conditioning systems. Initially, all driving trips are consolidated into a single CSV file, followed by identifying and removing Not a Number (NaN) values from the dataset. Due to seasonal and condition variations, inconsistent features with more missing values and outliers were identified and eliminated using recursive feature elimination (RFE). RFE efficiently selects the top fifteen features from the forty-seven, reducing computational costs, training time, and the risk of overfitting in high-dimensional datasets. This approach also mitigates the influence of feature inconsistency, making the model less susceptible to biases induced by seasonal changes and data discrepancies.

The LG 18650HG2 LIB dataset encompasses a range of ambient temperatures, including -10°C, 0°C, 10°C, and 25°C, capturing the battery's behaviour during charging, discharging, and pausing at each temperature setting. Various driving cycles, such as LA92, US06, UDDS, and HWFET, are used for battery evaluation, which simulates diverse real-world driving scenarios. Additionally, there are eight mixed cycles, denoted as mixed cycle 1 to mixed cycle 8, that are mixes of the four previously stated cycles, providing the battery with a broad spectrum of operating conditions. The model is trained with seven mixed cycles at ambient temperatures of -10°C, 0°C, 10°C, and 25°C.

The model was trained in a Windows 11 operating system environment, using an Intel i7-10510U CPU running at 1.80 GHz. The software stack included Python version 3.11.0, TensorFlow, Keras, and Jupyter Lab, providing a robust foundation for deep learning-based analysis and experimentation during model development. RMSE and MAE were chosen as evaluation metrics due to their simplicity, robustness to outliers, and interpretability. These metrics offer valuable insights into the model's performance and expected error levels, particularly when analyzing noisy and fluctuating time-series data from batteries. An early stopping strategy with a patience of 50 epochs was employed to prevent overfitting during training. The training process was set to iterate

Table 3.1: SOC Estimation for Different Driving Cycles.

Drive Cycle	Metrics	Ambient Temperatures			
		-10 Degree	**0 Degree**	**10 Degree**	**25 Degree**
LA92	MAE%	0.46	0.31	0.47	0.43
	RMSE%	0.79	0.40	0.65	0.52
UDDS	MAE%	0.45	0.32	0.62	0.51
	RMSE%	0.73	0.40	0.88	0.62
US06	MAE%	1.23	0.25	1.54	0.34
	RMSE%	1.82	0.37	1.88	0.45
Mix Cycle-7	MAE%	1.18	0.22	0.72	0.19
	RMSE%	1.65	0.32	0.86	0.24

for a maximum of 500 epochs, with a validation split of 0.2 used to monitor the model's performance on unseen data. The loss function used is the Huber loss function, which is highly effective in statistics and deep learning for robust regression. It handles outliers well by applying a linear treatment to significant errors, which helps manage their impact. The Huber loss function combines MAE and MSE in a balanced manner, improving optimization and enhancing overall model performance, making it a reliable choice for SOC estimation during training. The Huber loss function also effectively handles non-Gaussian noise resulting from varying operating conditions and environmental factors in battery data. The equation for the Huber loss function used in training the proposed model is given below:

$$\text{H-loss}(t) = \begin{cases} \frac{1}{2}(soc(t) - \hat{soc}(t))^2, & \text{if} |soc(t) - \hat{soc}(t)| \leq \delta \\ \delta(|soc(t) - \hat{soc}(t)| - \frac{\delta}{2}), & \text{otherwise} \end{cases} \tag{3.9}$$

where $H - loss(t)$ is Huber loss at time t, $soc(t)$ is the actual SOC at time t, $\hat{soc}(t)$ is predicted SOC at time t. The δ is a parameter that controls the transition from quadratic loss to linear loss. The choice of the value for δ depends on the specific characteristics of the data and the desired trade-off between robustness and precision in SOC estimation.

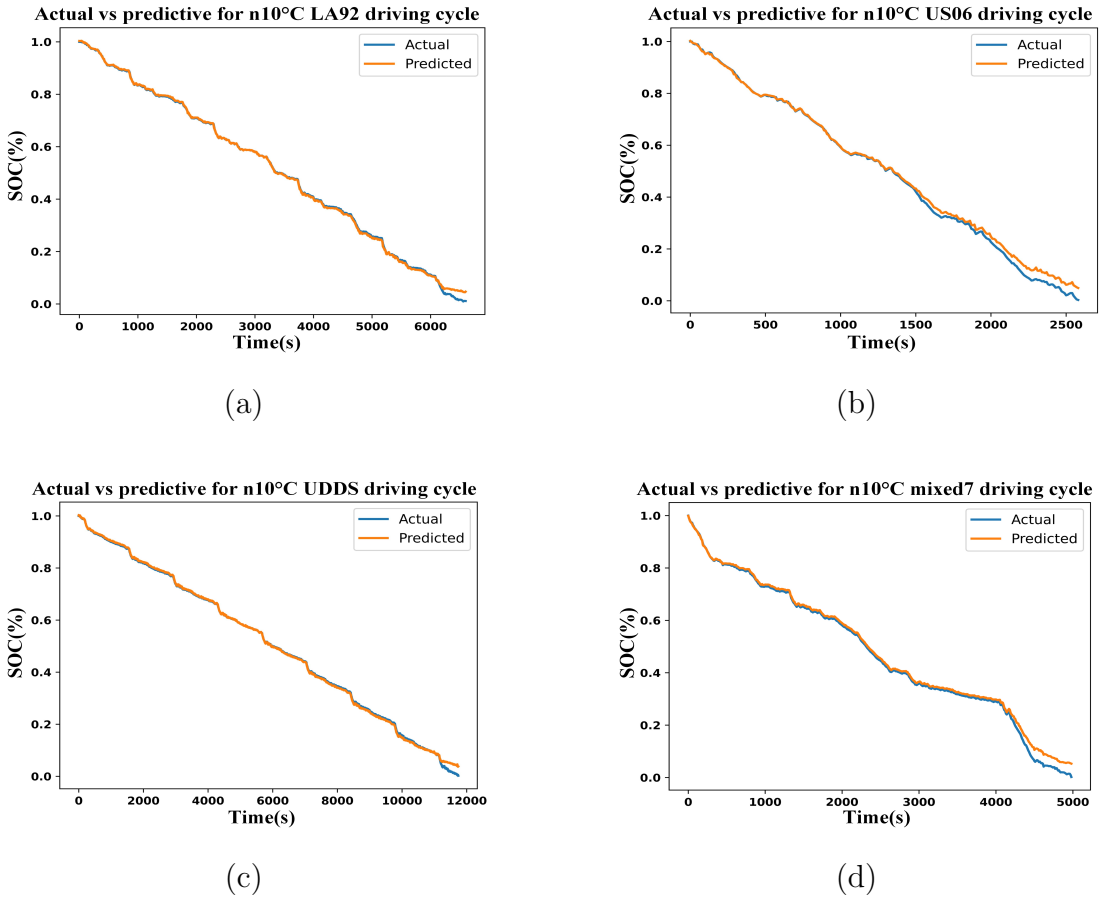


Figure 3.3: Estimation of SOC for different driving cycles at -10 degrees.

3.3.2 SOC Estimation Across Varied Driving Conditions and Ambient Temperatures

In the realm of EV battery management, the accurate estimation of SOC remains a paramount concern, especially under varying environmental conditions and diverse driving cycles. A novel approach leveraging the power of the MHDTCN-GRU network is used to precisely predict SOC levels, comprehensively evaluating SOC across multiple temperature profiles and driving cycles. The testbed encompasses established driving cycles such as LA92, US06, UDDS, and mixed cycle- 7. All the remaining driving cycles are used for training, and the entire architecture is trained for 500 epochs with a patience of 50 for early stopping.

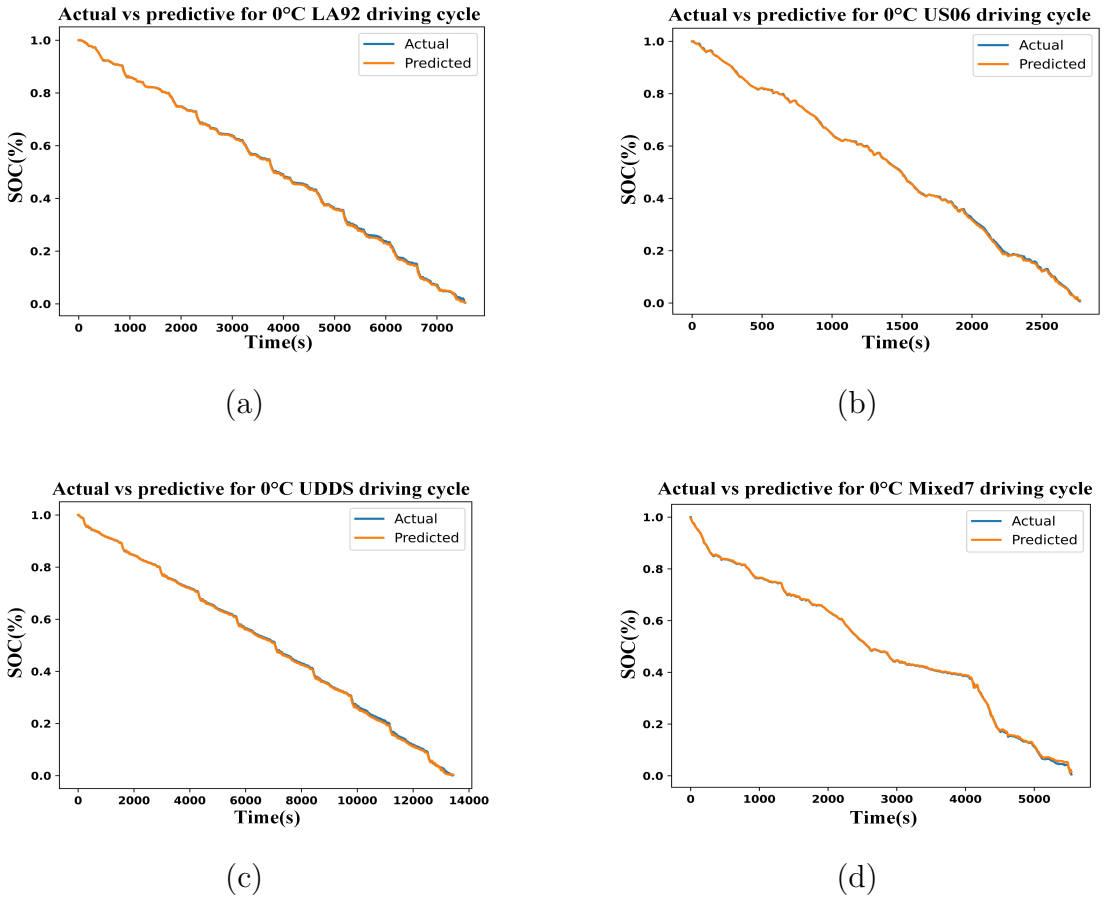


Figure 3.4: Estimation of SOC for different driving cycles at 0 degrees.

Table 3.2: Estimation under varying initial SOC values

Initial SOC	MAE	RMSE	Training Time
100%	0.0046	0.0072	2 hours 35 minutes 16 seconds
80%	0.0099	0.0122	2 hours 1 minute 47 seconds
60%	0.0098	0.0125	1 hour 25 minutes 36 seconds

The computational results highlight the proposed method's adaptability across diverse driving cycles and ambient temperatures. It achieves high estimation accuracy and robustness, with an MAE% of 0.54 and a RMSE% of 0.84. The hybrid model's efficient architecture, with only 374,433 parameters, indicates enhanced computational efficiency and reduced resource requirements, making it suitable for real-world applications in various environmental conditions. Table 3.1 presents the SOC estimation results for

various driving cycles at different temperatures. The hybrid model performs exceptionally well at 0 and 25 degrees, yielding low RMSE and MAE values. Fig. 3.3 shows the SOC estimation graphs with actual and predicted values for the general driving cycles LA92, UDDS, and US06, as well as the mixed driving cycle-mixed 7, all at an ambient temperature of -10 degrees. Similarly, Figs. 3.4, 3.5, and 3.6 display SOC estimation graphs for these driving cycles at 0 degrees, 10 degrees, and 25 degrees, respectively. Various architectural variants were explored by adjusting parameters and layers. The proposed model proved superior to alternative configurations by achieving lower RMSE and MAE values for SOC estimation.

To enhance the robustness of the proposed model and validate its applicability across diverse scenarios, it is essential to incorporate different SOC estimation intervals. The model's versatility can be thoroughly assessed by employing varying initial SOC values of 100%, 80%, and 60% across all temperature conditions, as shown in Table 3.2. Lower initial SOC values are associated with faster convergence times, with a notable decrease from 100% SOC (2 hours 35 minutes 16 seconds) to 80% SOC (2 hours 1 minute 47 seconds) and further to 60% SOC (1 hour 25 minutes 36 seconds). Conversely, higher initial SOC values yield better model accuracy initially, as evidenced by the superior MAE and RMSE values observed in the 100% SOC cases compared to the 80% and 60% SOC cases. However, a trade-off is apparent, as decreasing initial SOC values can improve convergence speed at the potential cost of accuracy. Nevertheless, the model demonstrates robustness across all tested SOC variations, converging with good results in each case.

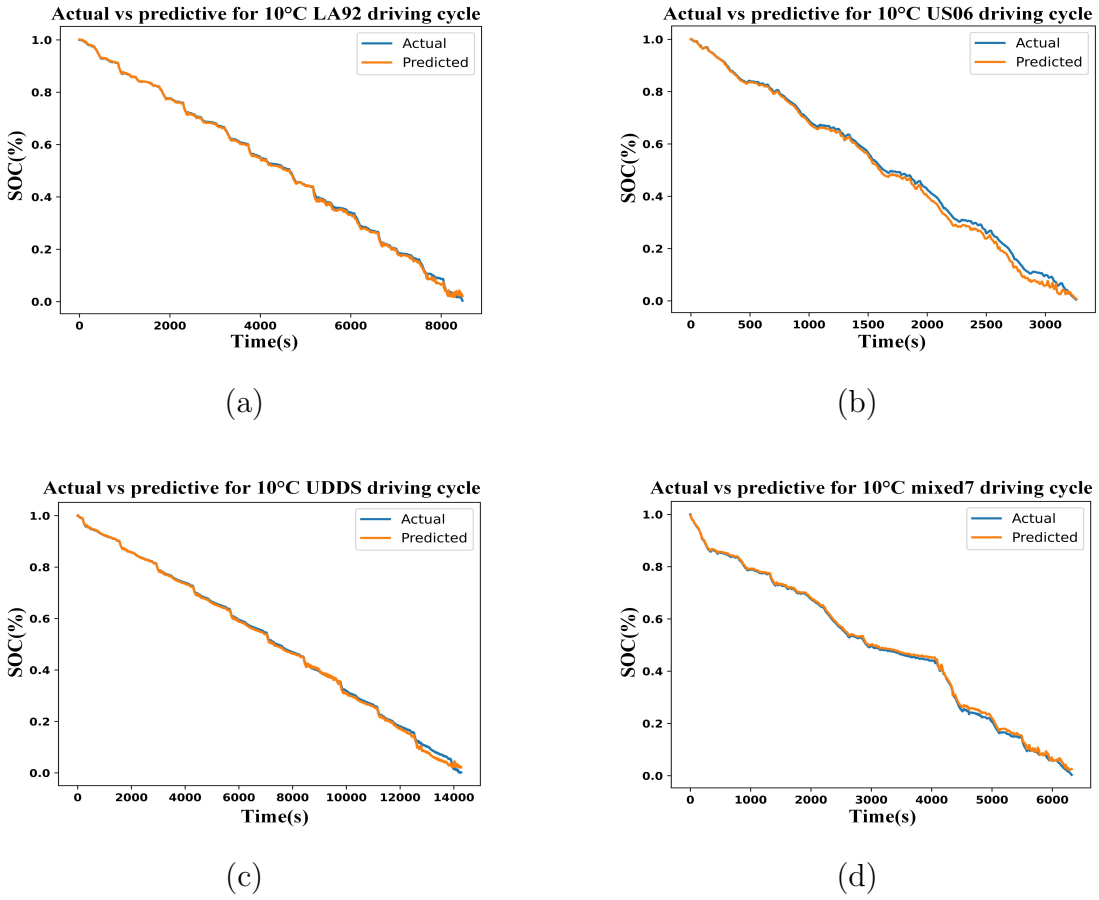


Figure 3.5: Estimation of SOC for different driving cycles at 10 degrees.

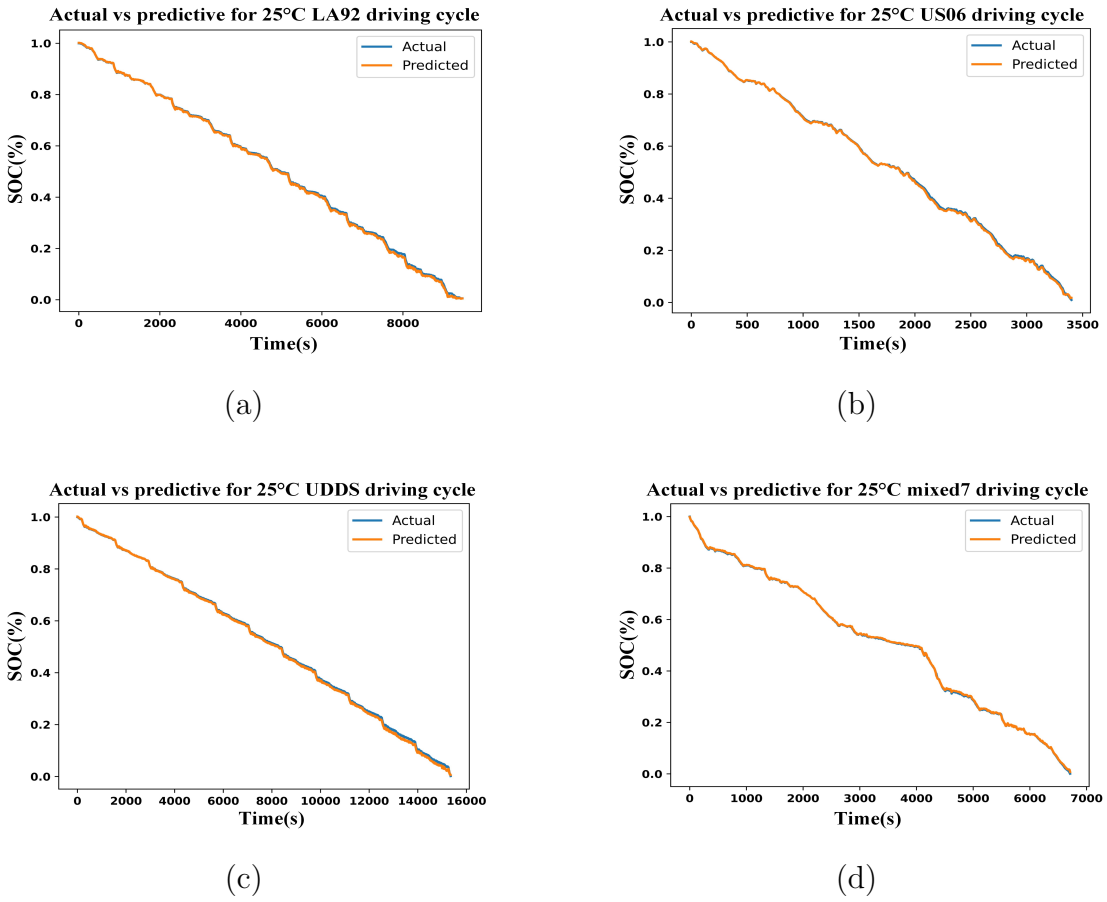


Figure 3.6: Estimation of SOC for different driving cycles at 25 degrees.

Table 3.3: SOC estimation results in LG 18650HG2 Li-ion Battery Data at different ambient temperatures

Methodology	Metrics	-10 degree Temperature	0 Degree Temperature	10 Degree Temperature	25 Degree Temperature	Parameters	Runtime
VMD+TCN [125]	MAE%	1.02	4.05	3.04	2.11	57313	2 hr 21 min
	RMSE%	1.25	5.23	3.88	2.83		
iBiGRU-UKF [126]	MAE%	-	0.83	**0.67**	0.52	-	-
	RMSE%	-	1.12	**0.74**	0.61		
Stacked-GRU [127]	MAE%	4.78	2.79	2.12	1.38	23601	53 min
	RMSE%	6.09	3.56	2.83	1.79		
CNN+BWGRU [128]	MAE%	2.55	0.52	0.69	0.52	555079	2 hr 11 min
	RMSE%	3.60	0.71	0.91	0.64		
PSO+TCN+Attention [129]	MAE%	0.90	0.58	0.85	0.63	56035	1 hr 15 min
	RMSE%	1.16	0.64	1.04	0.71		
CNN+BiLSTM+Attention [130]	MAE%	1.25	0.54	0.80	0.62	509183	3 hr 12 min
	RMSE%	1.51	0.64	1.02	0.78		
MHDTCN+GRU (Our Method)	MAE%	**0.88**	**0.31**	0.66	**0.42**	374433	1 hr 49 min
	RMSE%	**1.12**	**0.42**	0.88	**0.53**		

3.3.3 Comparative Analysis with Existing Approaches

In assessing the effectiveness of the proposed model for SOC estimation, state-of-the-art models were used for comparison. These models include cutting-edge algorithms such as variational mode decomposition (VMD) integrated with TCN, improved bidirectional gated recurrent unit (iBiGRU) networks with unscented Kalman filtering (UKF), and stacked GRU. Additionally, CNN combined with bidirectional weighted gated recurrent units (CNN+BWGRU) was employed, as well as a hybrid model incorporating particle swarm optimization (PSO) with TCN and attention mechanisms. Another model utilized CNN paired with bidirectional long short-term memory (BiLSTM) and attention mechanisms. To ensure consistency and comparability, the parameters for all these models were meticulously calibrated to match those specified in papers that utilize the same datasets. These are referenced in the accompanying Table 3.3. However, the iBiGRU-UKF model specified in [126] lacks explicit information regarding its dimensions and the number of its parameters.

The proposed model demonstrated notable improvements in RMSE% and MAE% values across various temperatures: 0.88 and 1.12 at -10°C, 0.31 and 0.42 at 0°C, and 0.42 and 0.53 at 25°C, respectively. However, at 10°C, the iBiGRU network with UKF exhibited better performance with an MAE% of 0.67 and RMSE% of 0.74. This parameter count in the proposed system is notably lower than other hybrid methods such as CNN-BiLSTM-Attention, which has a parameter count of 5,09,183, and CNN-BWGRU with 5,55,079 parameters. Despite this moderate parameter count, the proposed MHDTCN-GRU model achieves better results at -10 degrees, 0 degrees, and 25 degrees. The MHDTCN-GRU model maintains computational efficiency, completing the SOC estimation task in just 1 hour and 49 minutes—faster than several hybrid methodologies, including CNN-BiLSTM-Attention, CNN-BWGRU, and VMD-TCN. Although not the absolute fastest, the MHDTCN-GRU model achieves a balance of accuracy and efficiency, outperforming other approaches with reduced MAE and RMSE values. The results highlight the efficacy

of incorporating an MHDTCN with a GRU architecture. The combination delivered significantly improved accuracy and robustness in SOC estimation, underscoring its potential for real-time BMS. This hybrid design efficiently captures both the short-term and long-term temporal dependencies, thereby quickly adapting to learning the patterns and dependencies within the battery data. Furthermore, it does this with reduced parameter complexity.

The generalization of the proposed model is assured by evaluating the performance on diverse real-world BMW i3 driving cycles. Table 3.4 presents a comparative analysis of the above-mentioned methodologies for SOC estimation in real-world data from the BMW i3. This data focuses on key performance metrics and parameter complexity. The MHDTCN-GRU model, with a parameter count of 7,393, is notably more streamlined than other methods, which range from 9,353 to 17,393 parameters. It demonstrates a lower RMSE% of 0.0059 and an MAE% of 0.0049. Lower parameter complexity not only signifies reduced computational burden but also suggests potential advantages in terms of training efficiency, inference speed, and model interpretability. It achieves the best prediction accuracy, while maintaining a low inference time of 0.23 seconds, making it suitable for real-time SOC estimation in EVs.

The MHDTCN-GRU model has been evaluated across several dimensions, demonstrating its robustness, accuracy, parameter efficiency, and computational speed. The model adapts well to various SOC estimation intervals, performing consistently with initial SOC values of 100%, 80%, and 60%, which is crucial for real-world fluctuating battery conditions. It shows substantial reductions in RMSE% and MAE% across different temperatures, outperforming state-of-the-art algorithms using the LG 18650HG2 Li-ion Battery Data and achieving lower error values in various driving cycles. Additionally, it performs better on the real-world BMW i3 dataset, highlighting its generalization capability. The MHDTCN-GRU model also has a significantly lower parameter count (7,393 parameters) in the BMW i3 dataset compared to other hybrid methods (9,353

Table 3.4: SOC estimation in BMW i3 real-world data

Methodology	MAE (%)	RMSE (%)	Parameters	Inference Time (s)
Stacked GRU	0.0072	0.0086	16,705	0.22
CNN-BWGRU	0.0094	0.0124	13,985	0.32
CNN-BiLSTM-Attention	0.019	0.0137	17,393	0.25
VMD-TCN	0.0101	0.0127	17,281	0.21
PSO-TCN-Attention	0.0084	0.010	9,353	0.29
MHDTCN-GRU	**0.0049**	**0.0059**	**7,393**	0.23

to 17,393 parameters), enhancing its computational efficiency and making it suitable for resource-constrained environments. Moreover, it completes SOC estimation in 1 hour and 49 minutes, faster than several hybrid methods, while balancing accuracy and efficiency. Overall, the MHDTCN-GRU model is a robust, accurate, parameter-efficient, and computationally efficient solution, ideal for diverse applications requiring convergence and high accuracy.

The choice of GRU over LSTM is justified based on the quantitative results in the real-world dataset of BMW i3. The MHDTCN-GRU model achieves lower error metrics, with an MAE of 0.0049 and RMSE of 0.0059, compared to the MHDTCN-LSTM, which has an MAE of 0.0071 and RMSE of 0.0093. Additionally, the GRU-based model has slightly fewer trainable parameters (7,393) than the LSTM-based model (7,850), resulting in reduced computational complexity and faster training. These results indicate that GRU not only provides better predictive accuracy but also offers improved efficiency, making it a more suitable choice for SOC estimation in this work.

3.4 Ablation study

An ablation study was conducted to evaluate the contribution of each component in the proposed hybrid architecture. The MHDTCN-only model captures multi-scale temporal dependencies and provides a strong baseline with an MAE of 0.0061% and RMSE of 0.0075%. Incorporating the GRU layer enhances sequence learning and refines temporal representations, improving performance to an MAE of 0.0049% and RMSE of 0.0059%. This demonstrates that the combination of MHDTCN and GRU is critical for

Table 3.5: Ablation Study of MHDTCN-GRU Components for SOC Estimation

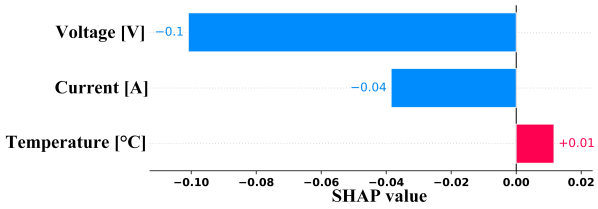
Methodology	MAE (%)	RMSE (%)	Parameters	Inference Time (s)
Stacked-GRU	0.0072	0.0086	16,705	0.22
MHDTCN	0.0061	0.0075	6,850	0.16
MHDTCN-GRU	**0.0049**	**0.0059**	**7,393**	0.23

accurate SOC estimation, highlighting the importance of both local feature extraction and sequential modeling in the hybrid framework. The ablation study is provided in 3.5.

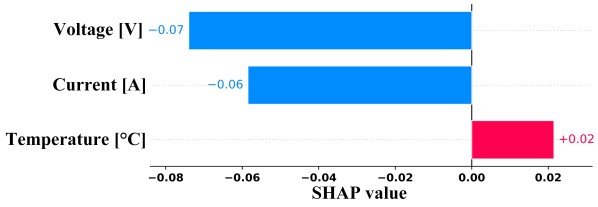
3.5 XAI for SOC estimation

The transparency of AI models and applications poses a persistent issue, hindering developers and consumers from understanding their inner workings. XAI is a solution to this problem that focuses on the AI model's ability to deliver human-understandable explanations for its judgments and predictions [131]. By making the decision-making process transparent and interpretable, XAI hopes to bridge the gap between human users and AI systems. SHAP values provide a consistent and theoretically supported method for attributing a model's output to its input attributes[132].

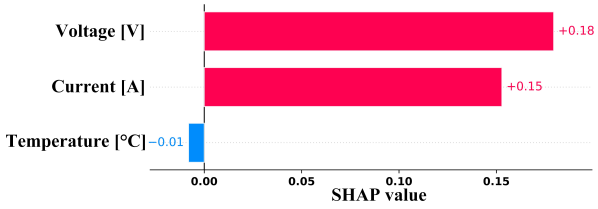
The bar plots in Fig. 3.7 indicate the feature importance of a single instance using SHAP. The red color indicates how much the input features affect the SOC prediction in the positive direction, and the blue indicates how much the input affects the SOC prediction in the negative direction. The bar plots for the negative ten-degree temperature, zero-degree temperature, ten-degree temperature, and twenty-five-degree temperature are shown in Fig. 3.7 (a), Fig. 3.7 (b), Fig. 3.7(c) and Fig. 3.7 (d) respectively. In negative ten temperature, current inversely impacts SOC the most, followed by voltage, with temperature playing a minor positive role. At zero degrees, voltage is the primary factor; both voltage and current negatively affect SOC, and temperature has a slight positive impact. At ten degrees, current has the highest positive impact, followed by voltage, with temperature slightly negatively affecting SOC. At 25 degrees, voltage has the most negative impact, while current positively influences



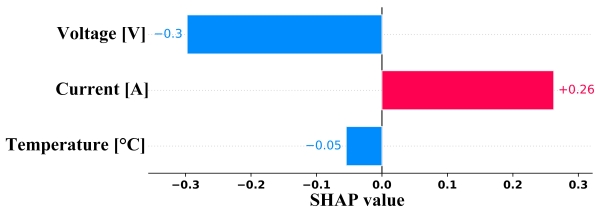
(a)



(b)



(c)



(d)

Figure 3.7: Feature importance of a single instance within the test data using SHAP.

SOC, with temperature having a minor negative effect. Feature importance fluctuates across varying temperatures for an individual instance, necessitating an overall analysis using summary plots for comprehensive insights across all test data samples.

A summary plot in Fig. 3.8 provides a visual representation of the impact of several features on the model's predictions. These plots extend the interpretation beyond a single sample, allowing us to discern consistent patterns and relationships between the features and the model's predictions across all the instances. If there are a lot of red

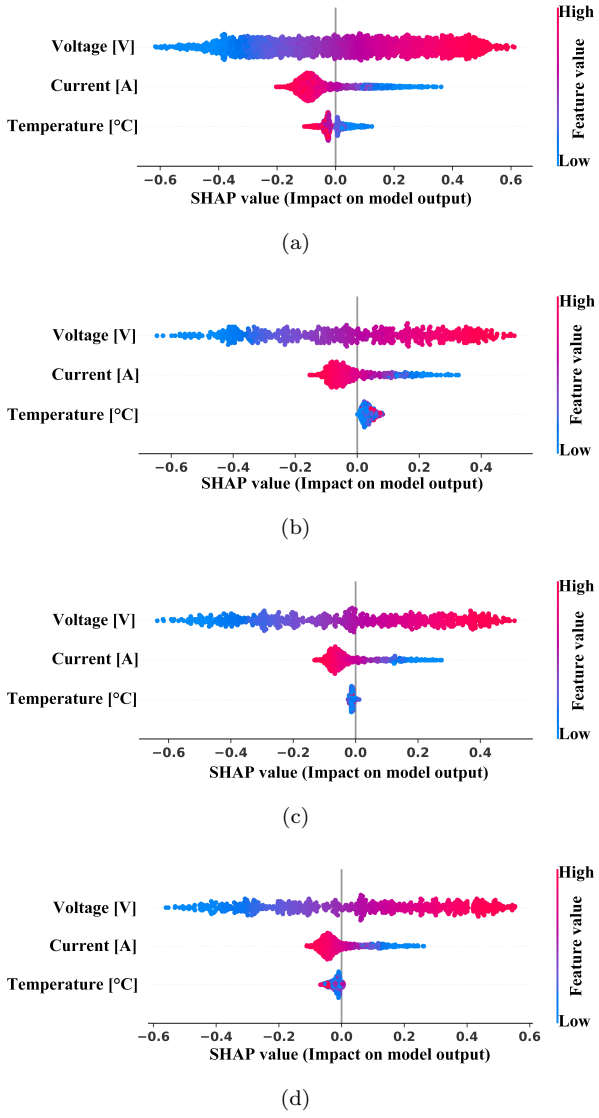


Figure 3.8: Feature importance of all the instances of test data using SHAP.

dots on the right half of the plot for a particular feature, it means that higher values of that feature have a positive impact on the output. Conversely, if there are a lot of blue dots on the left half of the plot for a feature, it indicates that lower values of that feature have a negative impact on the output.

The analysis of feature importance for all the instances conducted by SHAP at different ambient temperatures -10°C, 0°C, 10°C, and 25°C are shown in Fig. 3.8 (a), Fig. 3.8 (b), Fig. 3.8 (c), and Fig. 3.8 (d) respectively. Feature importance remains almost

consistent across varying temperatures for all the instances, clearly correlating input features and the predicted SOC for all the samples. The voltage of a battery emerges as the predominant factor influencing SOC estimation across diverse temperature conditions. A direct correlation exists between voltage fluctuations and predicted SOC levels: rising voltage indicates a higher SOC, while declining voltage signifies a lower SOC. This relationship is particularly evident at moderate temperatures, commonly around room temperature (25°C). Despite temperature variations, voltage remains a consistent and crucial indicator of SOC. However, it's important to note that the impact of voltage on SOC estimation may vary slightly based on temperature conditions and the specific characteristics of the battery chemistry.

The current is the second most significant factor affecting the estimation of SOC, typically showing an inverse relationship. Higher discharge currents tend to adversely affect the predicted SOC by hastening the depletion of stored energy, leading to a decrease in SOC. Conversely, lower discharge currents have a positive impact on predicted SOC, as they slow down the battery's energy depletion, thereby maintaining a higher SOC level. In colder temperatures (-10°C), the influence of current on SOC estimation may be more pronounced due to increased internal resistance and reduced ion mobility. However, with rising temperatures, the impact of current on SOC prediction might slightly diminish, reflecting the battery's enhanced performance and reduced sensitivity to current fluctuations within a warmer temperature range (10°C and 25°C).

The correlation between battery temperature and SOC can vary depending on the ambient temperature. At extremely low temperatures, such as -10°C, battery temperature and SOC have an inverse relationship. At 0°C, this inverse relationship still holds but with a less significant impact than -10°C, suggesting a slight reduction in the battery's sensitivity to temperature changes. As temperatures rise to 10°C and 25°C, the influence of battery temperature on SOC decreases further, indicating a more stable battery performance range. While the battery temperature still affects SOC within this

range, its impact is less pronounced, implying a more consistent SOC behavior with increasing temperatures.

The analysis utilizing SHAP values provided valuable insights into the relationship between the features (voltage, current, and temperature) and the predicted SOC of the battery. These findings can be instrumental in improving the transparency and interpretability of SOC prediction models and in enhancing the efficiency and longevity of battery systems in EVs.

3.6 Conclusion

This chapter presents a novel hybrid MHDTCN-GRU model for accurately estimating the SOC of battery in EVs at varying temperature conditions across multiple datasets. The contributions of the proposed work are, 1) The MHDTCN architecture effectively captures diverse data patterns by using multiple branches and concurrent processing with varied dilation rates. This enables the capture of long-term sequences in battery data; 2) Integration of GRU further enhances the model's performance by capturing intricate sequence patterns and temporal dependencies within battery data; 3) Comparative analysis demonstrates notable improvements in RMSE and MAE values, alongside a more compact parameter footprint, compared to existing hybrid models, and 4) The incorporation of XAI techniques, such as SHAP, offers transparent insights into SOC predictions, promoting trust and understanding of the model's decision-making process. These findings underscore the substantial potential of this hybrid model to advance SOC estimation accuracy, prioritizing interpretability and reliability for real-world applications in the realm of EV technology.

In summary, this chapter introduced a novel hybrid model which could precisely estimate SOC in LIBs with high robustness at various temperatures for different driving cycles and with computational efficiency. Although accurate SOC estimation is important in terms of real-time battery management and energy optimization, it is only a single

aspect of safe, and efficient battery operation. In practice, long-term performance and operational safety are equally governed by the battery's SOH. Thus, although SOC estimation provides insight into the battery's immediate usage, SOH prediction is critical for monitoring degradation patterns, planning maintenance, and extending the overall lifecycle. The next chapter builds on what we learned from SOC modeling by introducing a hybrid architecture for SOH prediction that uses multi-branch temporal convolutional structures with dynamic weight adaption, recurrent learning, and XAI. This extension from SOC to SOH allows the proposed research to cover both short-term operational accuracy and long-term reliability, providing a comprehensive framework for LIB management.