batteries

MDPI

Article

Lithium-Ion Battery Life Prediction Using Deep Transfer Learning

Wen Zhang [1], R. S. B. Pranav [1], Rui Wang [1], Cheonghwan Lee [2], Jie Zeng [1], Migyung Cho [3,*] and Jaesool Shim [1,*]

[1] School of Mechanical Engineering, Yeungnam University, Gyeongsan-si 38541, Republic of Korea; wenero777@yu.ac.kr (W.Z.); rsbpranav@yu.ac.kr (R.S.B.P.); wangrui20027@yu.ac.kr (R.W.); zengjie@yu.ac.kr (J.Z.)
[2] Korea Textile Machinery Convergence Research Institute, 27, Sampung-ro, Gyeongsan-si 38542, Republic of Korea; chlee@kotmi.re.kr
[3] Department of Computer & Media Engineering, Tongmyong University, Busan 48520, Republic of Korea
[*] Correspondence: mgcho@tu.ac.kr (M.C.); jshim@ynu.ac.kr (J.S.)

Abstract: Lithium-ion batteries are critical components of various advanced devices, including electric vehicles, drones, and medical equipment. However, their performance degrades over time, and unexpected failures or discharges can lead to abrupt operational interruptions. Therefore, accurate prediction of the remaining useful life is essential to ensure device safety and reliability. Conventional RUL prediction methods typically rely on regression analysis, signal processing, and machine learning techniques to assess battery conditions such as charge/discharge cycles, voltage, temperature, and durability. Although effective, these approaches are constrained by their dependence on large amounts of labeled data and the necessity for complex feature engineering to capture battery physical characteristics. In this study, we propose an approach that employs deep transfer learning to address these limitations. By leveraging pretrained model weights, the proposed method significantly improves the efficiency and accuracy of RUL prediction even under limited training data conditions. Furthermore, we investigate the impact of external environmental factors and physical battery characteristics on RUL prediction precision, thereby contributing to a more robust and reliable prediction framework.

Keywords: deep transfer learning; vgg16; lithium-ion battery; remaining useful life; prediction model

check for updates

Citation: Zhang, W.; Pranav, R.S.B.; Wang, R.; Lee, C.; Zeng, J.; Cho, M.; Shim, J. Lithium-Ion Battery Life Prediction Using Deep Transfer Learning. *Batteries* **2024**, *10*, 434. https://doi.org/10.3390/batteries10120434

Academic Editors: King Jet Tseng and Davy Cheong

Received: 31 October 2024
Revised: 2 December 2024
Accepted: 4 December 2024
Published: 6 December 2024

1. Introduction

In response to growing concerns about environmental pollution and energy shortages, focusing on renewable energy sources has become a priority across various sectors [1,2]. Lithium-ion batteries (LIBs), as outstanding energy storage devices, offer several advantages, such as high energy density, long lifespan, and low self-discharge rate, making them widely applicable in electric vehicles, consumer electronics, aerospace, and other advanced devices. However, despite these benefits, LIBs inevitably suffer from performance degradation and reduced lifespan during use. Therefore, accurately predicting the remaining useful life (RUL) of LIBs is crucial for ensuring their stability and safety.

In this context, many researchers have employed the concept of RUL to quantify the number of remaining charging and discharging cycles a LIB can undergo from its current state until it reaches the end of its useful life [3]. RUL is calculated as follows:

$$RUL = N_{EOL} - N_{CC}, \tag{1}$$

where N_{EOL} denotes the number of charge and discharge cycles the battery experiences when reaching the end-of-life; Ncc denotes the number of charge and discharge cycles the battery experiences at its current capacity.

To estimate the RUL of LIBs, researchers primarily use model-based approaches or data-driven methods. Model-based approaches encompass several categories, including equivalent circuit models [4], electrochemical models [5], empirical models [6,7], and physical models [8]. However, these methods are typically complex, with internal parameters that are difficult to obtain and simulate accurately under real-life conditions, which limits their practical application [9]. Wen et al. [10] proposed a model fusion scheme based on Physics-Informed Neural Networks (PINNs). This approach captures the degradation trend in battery State of Health (SOH) by developing semi-empirical, semi-physical partial differential equations (PDEs) to simulate the degradation dynamics of lithium-ion batteries. However, the general performance of PINNs may fall short in learning complex physical phenomena. Data-driven methods circumvent the reliance of conventional model-based prediction techniques on the internal parameters and aging mechanism of batteries [11]. Instead, they primarily use real-time data throughout the prediction process for comprehensive analysis, thereby ensuring accuracy without the need to construct a battery model. This approach reduces the influence of battery models on prediction results, thereby enhancing their applicability across various battery types, such as pouch, coin cells, and solid-state batteries [12]. In addition, data-driven methods excel at handling complex nonlinear patterns and relationships [13]. Deep neural networks (DNNs) [14], support vector machines [15], and Gaussian process regression [16] are prominent examples. Zhang et al. [17] developed a Gaussian process (GP) model with an exponential variance kernel to predict and classify battery lifespan. The model's performance was validated using battery datasets from three types of batteries under mixed operating conditions. Among these, deep learning (DL) methods, such as DNNs, have garnered significant attention in recent years due to their ability to automatically extract features, manage large datasets, and adapt flexibly to various scenarios. For example, Joonki Hong et al. [18] proposed a complete end-to-end DL framework to predict battery life. Although their framework can rapidly estimate the RUL of LIBs, limited experimental data may lead to a lack of intersection between the test and training datasets, resulting in inaccurate predictions. Wanjie Zhao et al. [19] proposed a DL model called CNN-AT to predict battery life, incorporating an attention mechanism into its feature extractor. Their model uses data from the early discharge phase of the battery and selectively highlights important features while suppressing irrelevant features in both the channel and spatial dimensions. However, the attention mechanism's need to capture detailed features poses a risk of overfitting, causing the model to perform well on training data but poorly on validation or test sets. Mingqiang Lin et al. [20] introduced a degradation trajectory prediction method based on synthetic data and DL. Their method successfully predicted the degradation trajectories of more than 100 LIBs during the early cycling stages using data from the full life of four cells. However, their method is limited to predicting early stages and cannot provide insights into the entire lifecycle.

In this paper, we propose an efficient method for predicting LIB lifetime using a deep transfer learning (DTL) model. The proposed method employs the VGG16 architecture as the convolutional neural network (CNN) backbone, leveraging diverse datasets, including current, voltage, power, and capacity, derived from battery aging and load experiments. These datasets are meticulously formatted for DTL input, allowing the convolutional layers to efficiently extract high-level features from data representations. Subsequently, the DTL model processes these features to yield accurate LIB lifetime predictions.

The primary contributions of this study can be summarized as follows:

We developed an innovative DTL model based on CNNs that is specifically tailored to predict LIB lifetime. The proposed model facilitates analysis and extraction of real-time battery parameters, enabling accurate predictions without relying on conventional internal battery data, structural models, or electrochemical reactions.

By integrating a wide range of data from battery aging and load experiments, the proposed approach allows for comprehensive comparative analysis, which improves prediction accuracy. The prediction outcomes are quantitatively assessed to identify the optimal model for battery lifetime estimation.

Currently, most existing studies have concentrated on battery cycles numbering fewer than 1500, often neglecting the crucial end-of-life stage of battery performance. Consequently, few research reports address the RUL prediction of lithium-ion batteries under extended cycling conditions [21–23]. This study takes a novel approach by generating a graphical representation for each cycle, producing 3000 images per battery. Using data collected from eight batteries, the dataset comprises 24,000 images, serving as the primary input for training and validating the proposed model. Through a rigorous testing framework, this research demonstrates the applicability of the proposed model across all stages of battery life, including the initial, optimal, declining, and end-of-life phases. This comprehensive methodology highlights the originality and robustness of the findings, offering valuable contributions to the field of battery health prediction and advancing the understanding of battery RUL under long cycling conditions.

2. Theories

2.1. Convolutional Neural Network (CNN)

CNNs are DL models, and their main feature is to extract the feature information of the local region of the input information layer by layer through the convolution kernel in the convolutional layer to capture the correlation between continuous data [24,25]. CNNs can reduce the number of network parameters by sharing the parameters of the convolutional kernel, which reduces the computational load and model size. The performance of CNNs can be improved by extracting different features between data with different convolutional kernel sizes and alternating them with pooling layers. CNNs are primarily composed of input, convolutional, pooling, fully connected, and output layers. The basic CNN structure is shown in Figure 1.

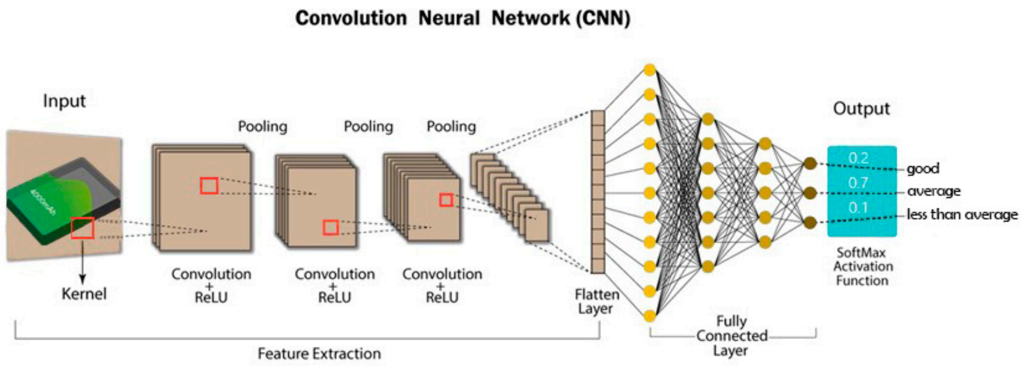

Figure 1. Schematic of convolutional neural network (CNN) structure.

The filter in the convolutional layer performs feature extraction using a suitable window and sliding step size. After pooling, the layer averages the processing of the high-level feature output to the fully connected layer to perform prediction or classification. The convolutional layer is calculated as follows [26]:

$$y_z^l = f\left(\sum\nolimits_{i\epsilon w_z} x_i^{l-1} \times k_{iz}^l + c_z^l\right) \tag{2}$$

where y_z^l denotes the z feature vector of the l layer of the output layer, f represents the activation function, w_z denotes the set of z features,x_i^{l-1} denotes the i value in the output of the l − 1 convolutional layer, k_{iz}^l denotes the i-th value in the z-th convolutional kernel of the l convolutional layer, and c_z^l denotes the bias of the convolution kernel of the lth convolution layer.

The pooling layer primarily down-samples the feature map to reduce feature dimensionality and the feature map size by combining the feature values in a certain interval into a single feature. A pooling layer typically employs maximum or average pooling. The maximum pooling used in this study can be expressed as follows:

$$y_j^l = \boldsymbol{Maxpooling}\left(x_j^{l-1}\right) \tag{3}$$

where y_j^l denotes the output of the lth pooling layer, the $Maxpooling$ max pooling layer function, and x_j^{l-1} denotes the output of the l −1th convolutional layer.

2.2. VGG16 Network

VGG16 is a type of typical CNN architecture. VGG16 was originally proposed by the Visual Geometry Group at the University of Oxford in 2014 [27]. VGG16 performs well in many tasks and is popular for its simple yet effective design. Its network structure is depicted in Figure 2a. Among the main features of VGG16 are that it has 16 layers of weights (13 convolutional layers and 3 fully connected layers) and uses multiple 3 × 3 convolutional kernels for feature extraction, which improves its expressiveness by decreasing the number of parameters and increasing the depth of the network compared to conventional large convolutional kernels.

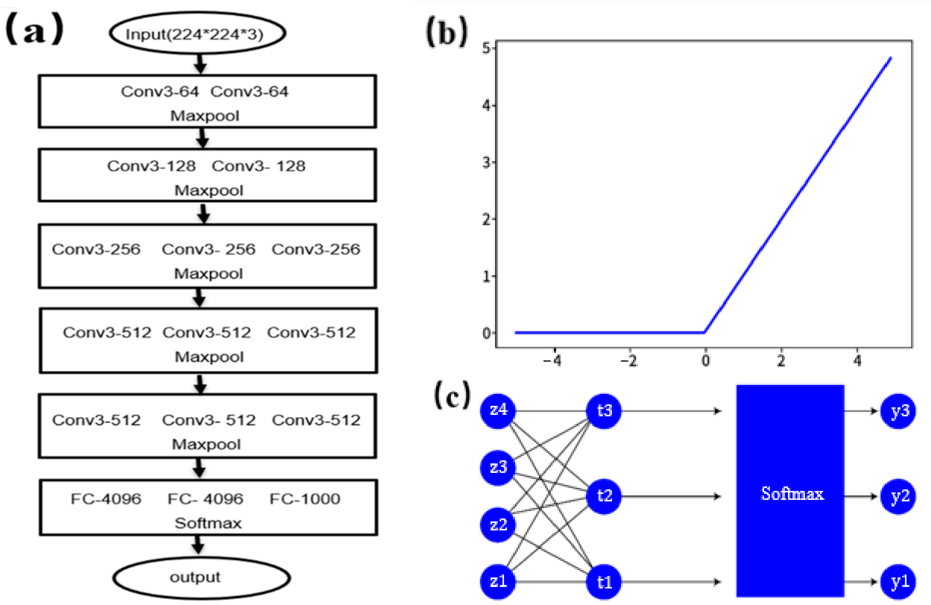

Figure 2. Components of predictive model: (**a**) VGG16 network structure, (**b**) ReLU activation function, and (**c**) Softmax activation function.

VGG16 follows the convolutional layer with a maximum pooling layer (2 × 2 pooling window), which is used to reduce the spatial dimensions of the feature map and computational effort. The fully connected layer is used to finalize the classification of features extracted from the convolutional layer. Behind the convolutional and fully connected layers, VGG16 employs a rectified linear unit (ReLU) activation function, which accelerates the training process and improves the nonlinear representation of the model. The ReLU activation function is illustrated in Figure 2b.

In the final layer of VGG16, classification is performed using the Softmax activation function, which is used to convert the raw outputs (logits) of the network into a probability distribution such that the predicted probability of each category sums to 1. The Softmax activation function is shown in Figure 2c. The Softmax activation function can be expressed as follows:

$$\boldsymbol{Softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}, \tag{4}$$

where z_i denotes the original output of the i category and $\sum_j e^{z_j}$ denotes the exponential sum of all categories used for normalization such that the predicted probabilities of all categories sum to 1.

3. Experiments and Method

This study involves three scenarios of battery aging tests and loading experiments. Figure 3a shows the setup for the battery aging experiment, which applies forced load from the battery aging device to rapidly cycle the battery through charge and discharge processes. Figure 3b illustrates the setup for the battery loading experiment, where the battery is discharged using an LED load in a real-world external environment until it is fully discharged, followed by recharging. Both experiments were conducted in a room-temperature environment, and the resulting LIB datasets were used for model performance validation [28].

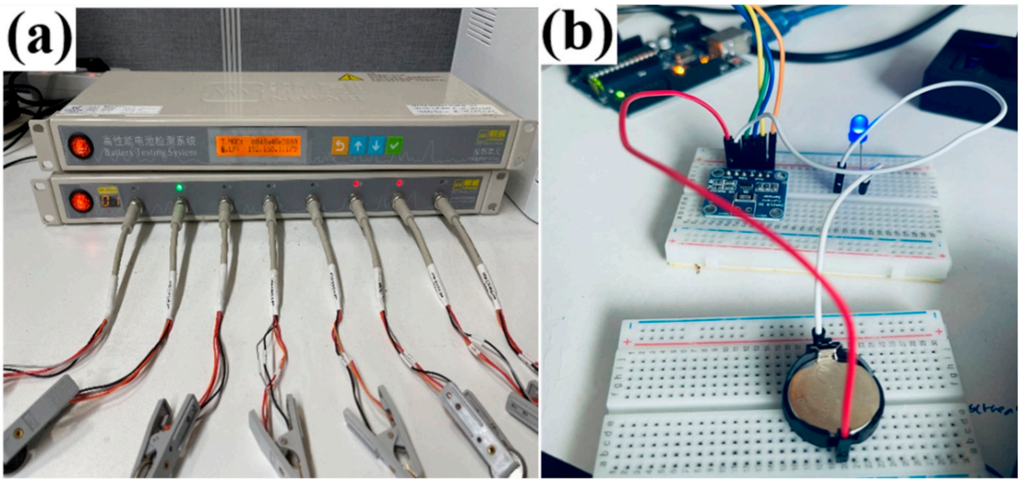

Figure 3. Battery test equipment: (**a**) battery aging and (**b**) battery load test equipment.

To ensure accuracy, comparability, and adherence to the experimental control variable method principles, the LIR2032 lithium-ion battery was selected for this study. As a widely commercialized and mature model in lithium-ion batteries, it has a rated capacity of 40 mAh. It has been reliably used in commercial applications for many years, making it an ideal and representative choice [29]. The experiments were conducted under room temperature conditions, with detailed specifications provided in Table 1.

Table 1. Battery parameters.

Clause	Parameter Values
Product name	EEMB LIR2032 (coin cell)
Rated capacity	40 mA
Rated voltage	3.6 V
Charging cutoff voltage	4.1 V
Discharging cutoff voltage	2.7 V
Charging current	35 mA
Discharging current	35 mA
Number of samples	8

In total, three datasets were used for prediction. The first and second datasets consisted of voltage–time and battery capacity data measured by the device during the battery aging test. The third dataset includes power, cutoff voltage, current, and time data recorded during the LED load experiment. The LED load experiment involved charging the battery with a constant current of 0.5 mA until a cutoff voltage of 4.1 V was reached. In the fully charged state, a 0.06 W LED was used as the load, and data were collected every 0.5 s until the battery was depleted. This cycle was repeated until the battery reached the end of 3000 cycles, marking the conclusion of the experiment. For the battery aging experiment, the battery was charged with a constant current of 35 mA to a cutoff voltage of 4.1 V and discharged with a constant current of 35 mA until the voltage dropped to 2.7 V. This process continued for 3000 cycles, after which the experiment was concluded. Figure 4 illustrates the capacity decay curve for batteries 1–8 during the battery aging test.

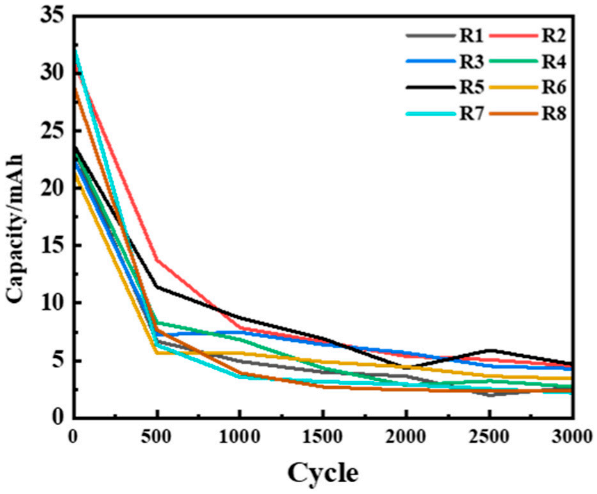

Figure 4. Capacity decay curves for batteries 1–8 in the battery aging test.

The experiments described in Figure 3 were conducted in a controlled laboratory environment to minimize the influence of external factors, including temperature. Specifically, both the battery aging and LED load experiments were performed at room temperature, consistently maintained within a range of 20 °C to 30 °C. This range was carefully monitored and controlled throughout the experiments to ensure uniform conditions. Given that numerous studies have reported that lithium-ion batteries perform optimally within a temperature range of 15 °C to 35 °C [30], we can conclude that the observed capacity decay was primarily a result of battery cycling rather than temperature effects.

Figure 5 shows the physical characteristics of a new battery over one cycle to examine its aging properties. Figure 5a,b show the voltage changes during charging and discharging for one cycle and the voltage–capacity data obtained from the battery aging test device. These figures reflect the aging characteristics of the batteries. Figure 5c shows the battery power, cutoff voltage, and current data acquired from the battery loading experiment setup. Note that only data related to battery discharge were considered in this case. The graphs obtained from 1 to 3000 cycles were used as input feature values for transfer learning. The selection of the voltage–time, voltage–capacity, and power data as the key features of the DTL model is based on the ability of these features to capture the fundamental aging characteristics of LIBs. Voltage and capacity provide direct insights into charge–discharge cycles, and power-related metrics reflect battery performance under a load, making these features essential for accurate RUL prediction.

To analyze the impact of external environmental conditions on the experiments, we conducted tests during summer and winter. Figure 6 illustrates the data variation of the three methods in the summer experiment. Figure 7 shows the data variations of the three methods in the winter experiment. In the graphs used as input features for transfer learning, the x- and y-axis labels were removed, and the model was trained using only the images.

By analyzing the data presented in Figures 6 and 7, we observe that the aging characteristics of Methods I and II are less sensitive to seasonal changes across different cycles. In contrast, Method III exhibits a pronounced sensitivity to temperature variations in terms of aging characteristics.

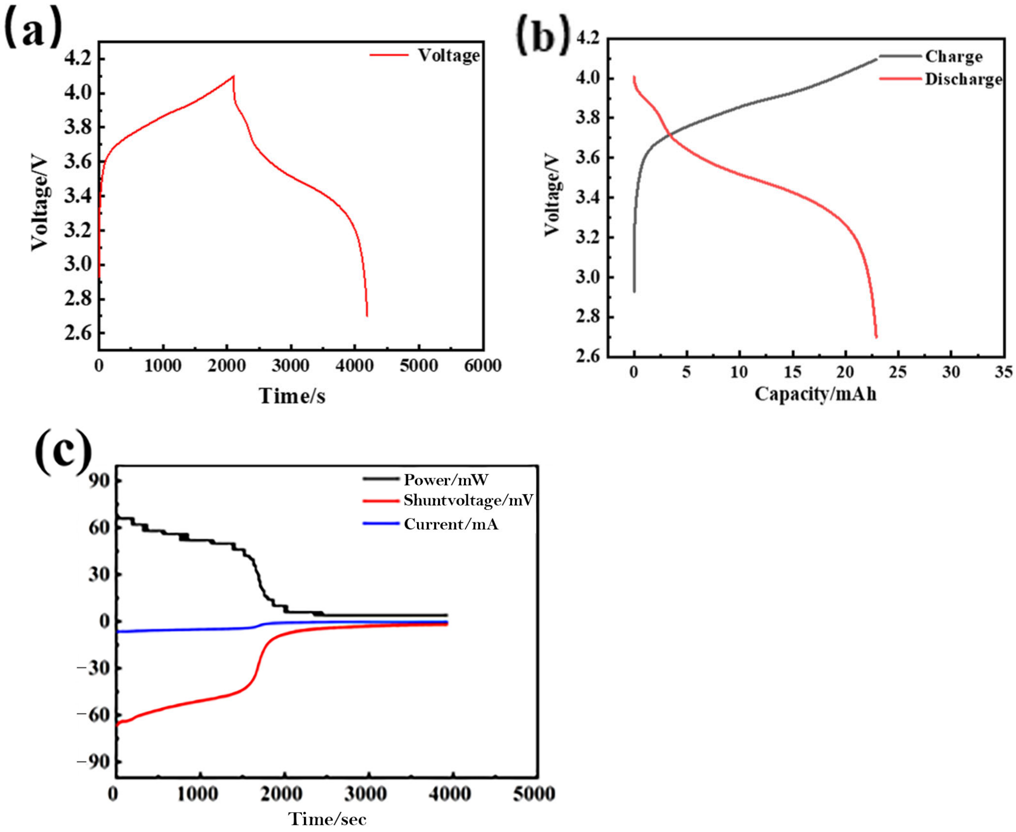

Figure 5. Experimental data: (**a**) Method I: voltage–time data from the battery aging experiment; (**b**) Method II: voltage–capacity data; (**c**) Method III: power–cutoff voltage–current–time data.

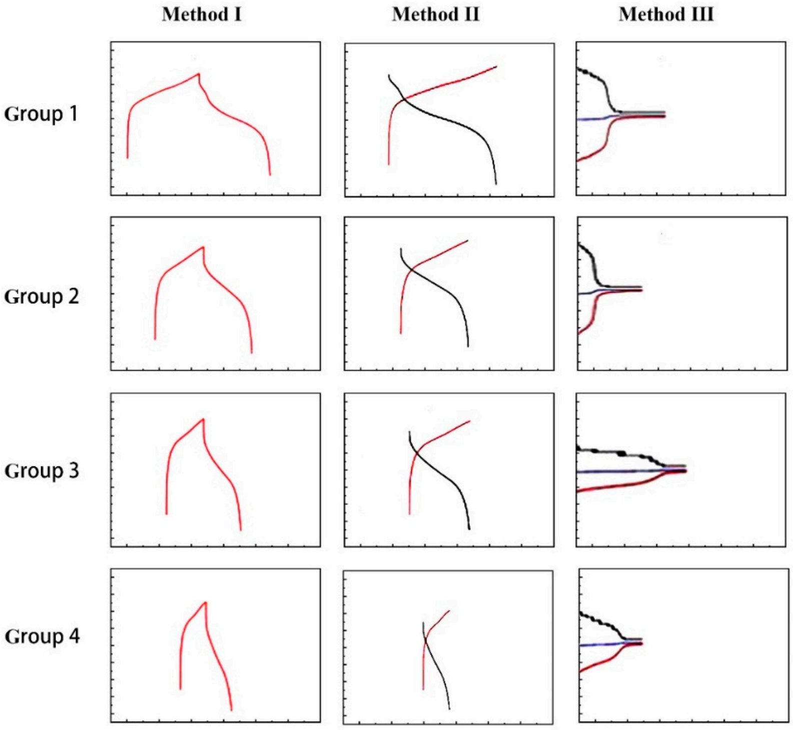

Figure 6. Feature variations over cycles tested during summer.

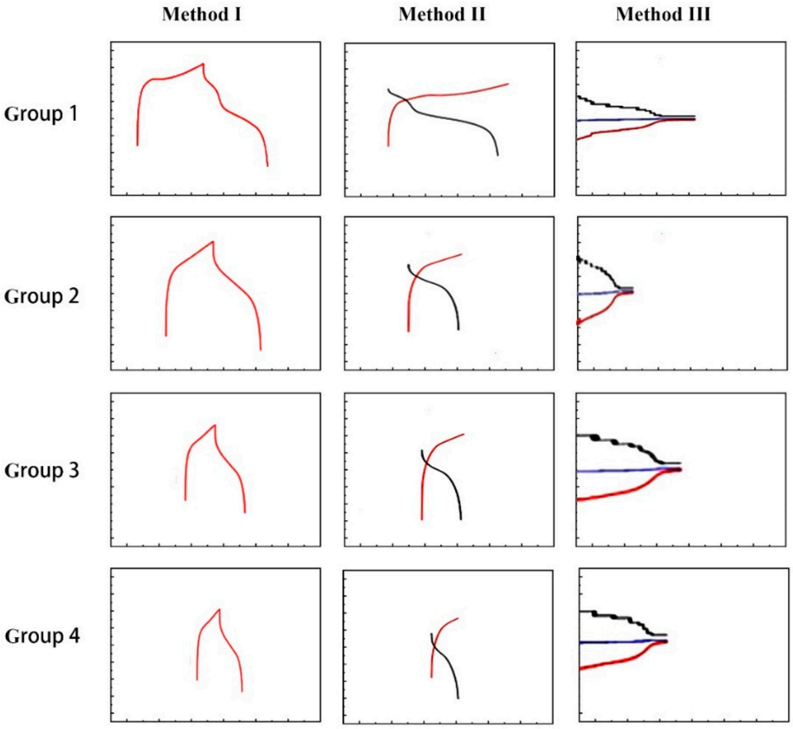

Figure 7. Feature variations over cycles tested during winter.

In addition to the influence of temperature, the nature of data-driven methods necessitates preprocessing raw cell data due to issues such as anomalies and inconsistencies encountered during data acquisition. This process involves handling outliers, duplicate data, and missing values and selecting the most appropriate battery data for each method.

In this study, data obtained from 8 samples over 3000 cycles, totaling 24,000 data points per method, were prepared for DTL. For 0 cycles, the data from 0 to 250 cycles were grouped as Group 1, 251 to 1000 cycles as Group 2, 1001 to 2,000 cycles as Group 3, and 2001 to 3000 cycles as Group 4.

3.1. DTL Model Building

Transfer learning is a machine learning technique that significantly improves model performance and training efficiency by leveraging the knowledge acquired from a task to improve learning in a related task. In this context, we use the VGG16 architecture, which preserves convolutional layers while removing fully connected layers. Instead, we introduce three fully connected layers, resulting in a total of seven output neurons.

Compared to conventional neural networks, CNNs excel at efficiently extracting high-level features from data while requiring fewer parameters for network construction and minimizing computational load to improve accuracy and speed. However, training CNNs can be resource-intensive, necessitating a substantial amount of labeled data and posing a risk of overfitting [31]. Transfer learning effectively leverages small datasets, which mitigates the risk of overfitting [32]. Therefore, this study integrates these two methodologies to establish a DTL network. The architecture of the DTL network is illustrated in Figure 8.

In our implementation, we set the hyperparameters as follows: the learning rate was set to 0.0001, the batch size was 64, and the Adam optimizer was employed. The number of training epochs was set to 100, and the first 12 layers of the VGG16 model were frozen to prevent updates during training, whereas the last three layers were trainable.

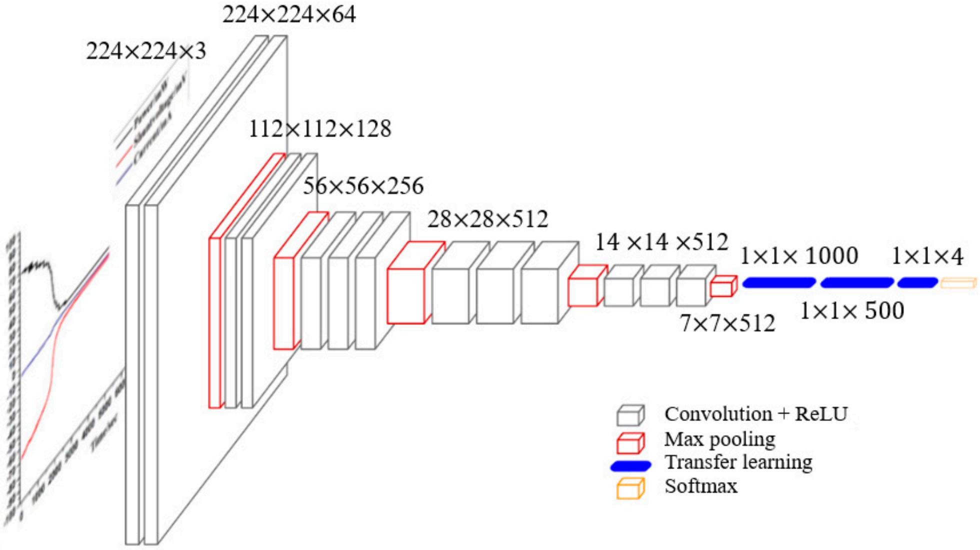

Figure 8. Schematic of deep transfer learning (DTL) network architecture.

The update rule of the Adam algorithm is outlined as follows:

i. Gradient calculation at time step t:

$$g_t = \nabla_{\theta} J(\theta_{t-1}). \tag{5}$$

ii. Update the exponential moving average of the gradient m$_t$ and the exponential moving average of the gradient squared v$_t$ as follows:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t, \tag{6}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2. \tag{7}$$

Here, β_1 denotes the first-order attenuation rate, β_2 denotes the second-order attenuation rate, and m_t and v_t are initialized with values of 0.

Initializing m_t and v_t with values of 0 causes m_t and v_t to be biased toward 0; thus, bias correction is required to reduce the impact of the bias on the initial training stage.

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}, \tag{8}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t}. \tag{9}$$

iii Update the parameters as follows:

$$\theta_t = \theta_{t-1} - \alpha \times \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon}, \tag{10}$$

where α denotes the initial learning rate and ε denotes the smoothing exponent.

In this study, we used the VGG16 neural network, which was trained on the source task while preserving the convolutional layers and discarding the learned fully connected layers. Instead, we introduced a series of layers following the flattening layer (Figure 1) consisting of 1000 dense units, followed by 500 dense units and four output categories: "like new," "excellent," "average," and "bad." In addition, we employed input images sized at (224, 224, 3) to facilitate compatibility with the VGG16 architecture.

3.2. Class Weighting

To mitigate the effect of data imbalance across different classes, we employed a class weighting scheme. Specifically, a weight wiw_iwi was assigned to each class iii, where wiw_iwi is inversely proportional to the number of samples (N_{class_i}) in the class, ensuring that less frequent classes are not under-represented in the training process. The weight for each class is calculated as follows:

$$\boldsymbol{\omega_i = \frac{N_{total}}{N_{class_i}}}, \tag{11}$$

where N_{total} denotes the total number of samples and N_{class_i} denotes the number of samples in class i. This weighting scheme ensures that the loss function gives equal importance to each class, which prevents bias toward more frequent classes. This can be easily implemented in TensorFlow using the class_weight parameter in the model.fit() function, which effectively accounts for class imbalance.

4. Results and Discussion

The processor used for the experimental hardware in this model was an Intel(R) Core (TM) i5-6300HQ CPU (2.30 GHz) with 8 GB RAM. The SSD capacity was 500 GB, the computer operating system was Windows 10 Pro for Workstations, and the model code was executed in the Visual Studio code program framework, version: 1.89.1; version number: Windows_NT ×64 10.0.19045. The LIB vector dataset was processed and transformed into a format suitable for the DTL model. Next, the dataset was split into a training set (80%) and a validation set (20%). The proposed model was trained and validated on the training and validation datasets, respectively. The experimental results are presented in Figure 9.

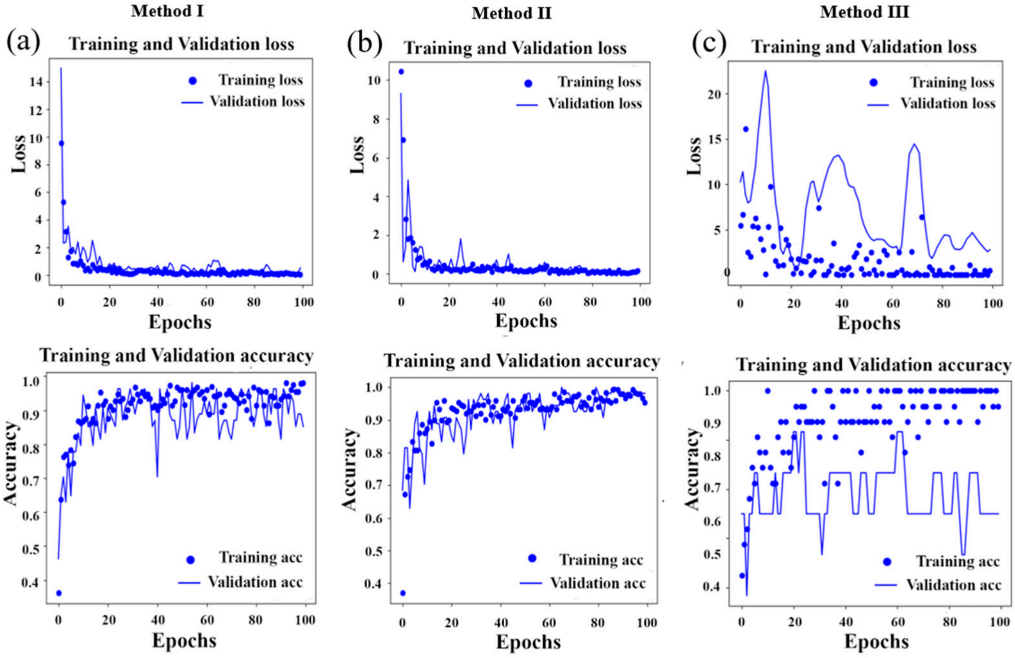

Figure 9. Training and validation losses: training and validation accuracy results obtained by DTL model. (**a**) Results of Method I; (**b**) results of Method II; and (**c**) Results of Method III.

Figure 10 shows the loss and accuracy values obtained during training and validation on the three datasets. The training loss decreases rapidly across all models, indicating that the transfer learning weights are well-fitted to the training data. However, the validation loss behaves differently for each method. Methods I and II exhibit relatively stable validation losses, indicating that these methods exhibit good generalizability on the validation dataset. In contrast, Method III exhibits more significant fluctuations in validation loss, with significantly higher values. This result suggests that Method III may have overfitted the data, leading to unstable performance on the validation data. This problem arises due to the variability in feature values caused by external factors, particularly temperature changes and cycle variations, which do not provide good training data.

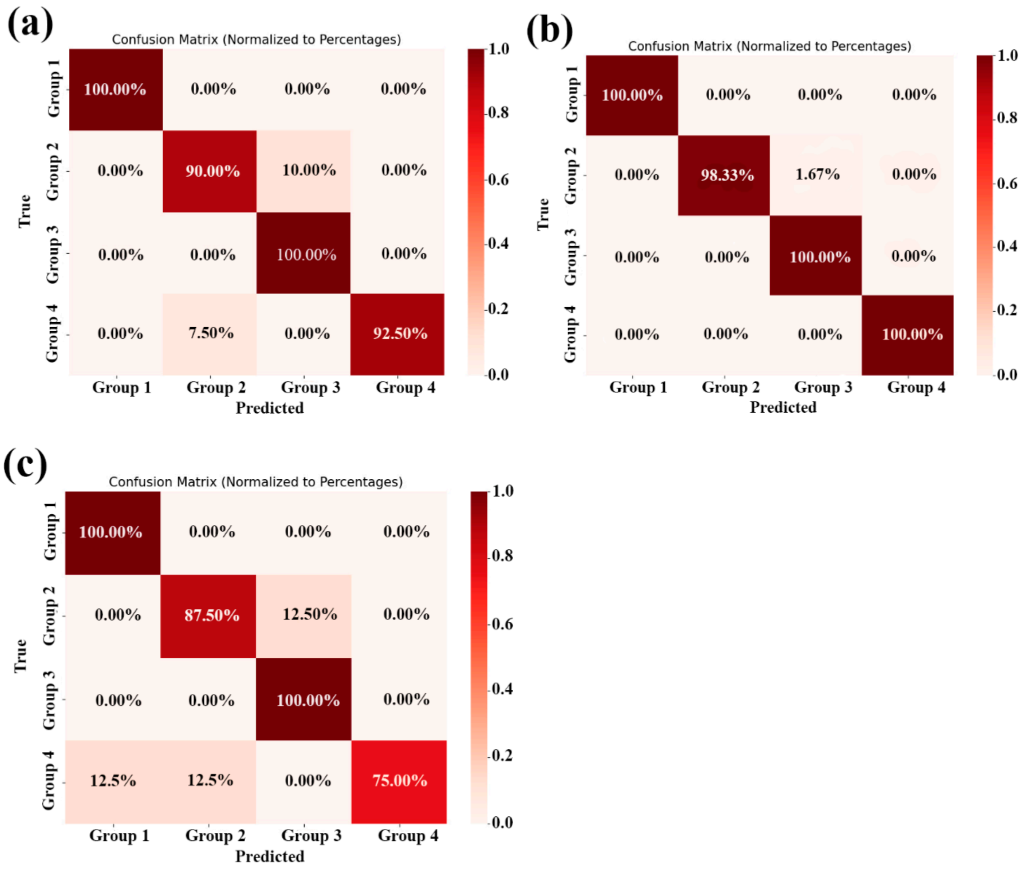

Figure 10. Confusion matrix. (**a**) Results of Method I, (**b**) Method II, and (**c**) Method III.

In terms of accuracy, Methods I and II significantly outperformed Method III. In particular, Method II exhibited continuous performance improvement as the number of epochs increased. For Method I, considering both voltage and capacity over one cycle provided a better representation of the battery aging characteristics than focusing on the voltage changes over time. A confusion matrix was also used to summarize the evaluation of the DTL model (Figure 10).

The performance of the DTL model was evaluated across Methods I, II, and III using key metrics, including accuracy, precision, recall, and F1-score. These metrics provide a comprehensive assessment of the classification performance and offer insights into the strengths and weaknesses of each method.

Accuracy is the proportion of correctly predicted instances, including both true positives and true negatives, out of the total number of predictions. It is calculated as the sum of true positives and true negatives divided by the total number of samples. Accuracy gives an overall measure of how often the model makes correct predictions.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

where TP is the true positive, TN is the true negative, FP is the false positive, and FN is the false negative.

Precision focuses on the reliability of positive predictions by calculating the proportion of true positives among all instances predicted as positive. This metric is particularly important when the cost of false positives is high, as it ensures that the positive predictions are accurate.

$$\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP} + \boldsymbol{FP}} \tag{13}$$

Recall, also known as sensitivity or the true positive rate, measures the model's ability to identify actual positive instances. It is calculated as the proportion of true positives out of all actual positive instances. Recall is critical when the consequences of missing positive instances (false negatives) are severe.

$$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP} + \boldsymbol{FN}} \tag{14}$$

F1-score is a metric that balances precision and recall. It is the harmonic mean of these two metrics, providing a single measure that accounts for both false positives and false negatives. This makes it especially useful for evaluating models on imbalanced datasets, where one class may dominate the others.

$$\boldsymbol{F1} - \boldsymbol{Score} = 2 \times \frac{\boldsymbol{Precision} \times \boldsymbol{Recall}}{\boldsymbol{Recision} + \boldsymbol{Recall}} \tag{15}$$

The performance of the model was analyzed using these metrics for each group and method. Below is a summary of the results.

From Table 2, it is evident that Method II outperforms the other methods across most metrics and groups. It achieves the highest accuracy, precision, recall, and F1-scores, reflecting its robustness and reliability in classifying battery states. Method I demonstrates stable performance, particularly in Groups 1 and 3, but exhibits minor misclassifications in Groups 2 and 4. On the other hand, Method III struggles significantly, particularly in distinguishing between Group 4 (representing bad batteries) and Group 1 (representing new batteries), resulting in lower precision, recall, and F1-scores.

Table 2. Performance metrics summary for Methods I, II, and III.

Method	Group	Accuracy	Precision	Recall	F1-Score
Method I	Group 1	100.0	100.0	100.0	100.0
Method I	Group 2	95.0	100.0	90.0	94.7
Method I	Group 3	100.0	100.0	100.0	100.0
Method I	Group 4	96.5	92.9	100.0	96.3
Method II	Group 1	100.0	100.0	100.0	100.0
Method II	Group 2	99.5	99.0	100.0	99.5
Method II	Group 3	100.0	100.0	100.0	100.0
Method II	Group 4	100.0	100.0	100.0	100.0
Method III	Group 1	100.0	100.0	100.0	100.0
Method III	Group 2	94.0	87.9	100.0	93.5
Method III	Group 3	100.0	100.0	100.0	100.0
Method III	Group 4	87.5	75.0	100.0	85.7

This comprehensive comparison underscores the strengths of Method II while identifying areas for improvement in Methods I and III, particularly in handling data variability and mitigating critical misclassifications.

The cycle-based average error was calculated using the confusion matrix by evaluating the differences between the actual and predicted groups. Each group corresponds to a specific cycle range, and the midpoint of that range was used to estimate errors when predictions deviated from the correct group. The calculation involved summing the absolute differences between the midpoints of the actual and predicted groups, weighted by the number of misclassifications, and dividing by the total number of samples in the group.

For this analysis, the cycle ranges and midpoints were defined as follows:

- **Group 1** corresponds to cycles 0–250, with a midpoint of 125.

- **Group 2** corresponds to cycles 251–1000, with a midpoint of 625.
- **Group 3** corresponds to cycles 1001–2000, with a midpoint of 1500.
- **Group 4** corresponds to cycles 2001–3000, with a midpoint of 2500.

The average error (±cycles) was calculated using the formula:

$$\mathrm{AE}(\pm\mathrm{Cycles}) = \mathrm{MR}(\%) \times \mid \mathrm{PM} - \mathrm{AM} \mid \tag{16}$$

Herein, AE means average error (±cycles), MR represents the misclassification rate, PM is the predicted midpoint, and AM means the actual midpoint.

In this study, we adopted this approach to provide a more intuitive understanding of classification performance. Misclassified samples were quantified based on the differences between the midpoints of the actual and predicted groups, enabling a clear interpretation of classification errors in terms of their impact on cycle predictions.

For Method I, Group 1 achieved perfect classification accuracy with no errors. Group 2 exhibited a 10% misclassification rate, with samples being misclassified as Group 3, leading to an average error of ±87.5 cycles. Similarly, Group 3 achieved perfect classification accuracy, consistent with Group 1, and showed no errors. Group 4, however, experienced a 7.5% misclassification rate to Group 3, resulting in an average error of ±75 cycles.

For Method II, the same calculation method was applied. Group 1 achieved perfect classification accuracy, resulting in no errors. Group 2 exhibited a 1.67% misclassification rate to Group 3, yielding an average error of ±14.6 cycles. Both Group 3 and Group 4 achieved perfect classification accuracy with no errors.

For Method III, Group 1 achieved perfect classification accuracy, resulting in no errors. Group 2 showed a 12.5% misclassification rate to Group 3, which resulted in an average error of ±109.4 cycles. Group 3 achieved perfect classification accuracy with no errors, while Group 4 experienced a 12.5% misclassification rate to Group 1 and another 12.5% to Group 2, leading to a total average error of ±531.3 cycles.

These results highlight that Method II demonstrated the most consistent and accurate performance across all groups, achieving minimal average errors. Method I showed moderate performance with reasonable accuracy but higher errors compared to Method II. Conversely, Method III faced challenges with higher misclassification rates and significantly larger average errors, particularly for Group 4, underscoring the need for improved handling of variability in data.

Table 3 presents a detailed comparison of classification methods based on cycle-based error analysis, showcasing their respective strengths and weaknesses. This analysis underscores Method II as the most consistent performer, offering a robust balance of accuracy and minimal error. By leveraging cycle-based metrics, this approach provides a clear and quantitative evaluation of classification performance in a practical and intuitive manner.

Table 3. Comparison of classification methods based on cycle-based error analysis.

Method	**Group**	**Cycle Range**	**PM (Cycles)**	**MR (%)**	**AE (±Cycles)**
Method I	Group 1	0–250	125	0	0
Method I	Group 2	251–1000	625	10	87.5
Method I	Group 3	1001–2000	1500	0	0
Method I	Group 4	2001–3000	2500	7.5	75.0
Method II	Group 1	0–250	125	0	0
Method II	Group 2	251–1000	625	1.67	14.6
Method II	Group 3	1001–2000	1500	0	0
Method II	Group 4	2001–3000	2500	0	0
Method III	Group 1	0–250	125	0	0
Method III	Group 2	251–1000	625	12.5	109.4
Method III	Group 3	1001–2000	1500	0	0
Method III	Group 4	2001–3000	2500	25	531.3

5. Conclusions

This study evaluated three methods, Methods I, II, and III, using a DTL model to predict the RUL of LIBs. Among the methods, Method II demonstrated the best performance in terms of both accuracy and generalizability due to its effective feature extraction using the VGG16 architecture. By incorporating voltage and capacity data, Method II provided the most accurate representation of battery aging characteristics and maintained stable performance even under varying environmental conditions. In contrast, Method III exhibited significant sensitivity to external factors, particularly temperature variations and cycle changes, leading to higher validation loss fluctuations and misclassification issues. Notably, Method III frequently misclassified bad batteries (Group 4) as new batteries (Group 1), highlighting the need for further refinement in feature extraction and data processing to improve robustness. Method I exhibited stable performance with minor classification issues but was generally outperformed by Method II. These results emphasize the importance of including comprehensive data features, such as voltage and capacity, to better capture LIB aging dynamics. Overall, the proposed DTL model proved to be highly effective for battery life prediction, particularly when robust feature extraction and preprocessing methods were applied. Future work focuses on addressing improved feature extraction techniques and optimizing experimental setups to minimize the effects of external variations and improve model robustness, especially for methods such as Method III, as well as predicting the remaining lifetime of Li-ion batteries using other approaches such as differential voltage analysis methods and EIS measurements of impedance. We plan to incorporate these features in future studies and conduct experiments using larger datasets to further enhance the generalizability of our model.

Author Contributions: W.Z.: visualization, methodology, writing—original draft. R.S.B.P.: software. R.W.: writing—review and editing. C.L.: validation. J.Z.: investigation. M.C.: investigation, formal analysis. J.S.: conceptualization, supervision, funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by a National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS2023-00280665), a Korea Institute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Korea government (MOTIE) (RS2024-00451324), the Technology Innovation Program funded by the Ministry of Trade Industry & Energy (MOTIE, Republic of Korea) (20023578 and RS2023-00246529), and the Technology development Program (S3366561) funded by the Ministry of SMEs and Startups (MSS, Republic of Korea).

Data Availability Statement: The data that support the findings of this study are available upon request from the corresponding author.

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

References

1. Wang, R.; Jang, W.Y.; Zhang, W.; Reddy, C.V.; Kakarla, R.R.; Li, C.; Gupta, V.K.; Shim, J.; Aminabhavi, T.M. Emerging two-dimensional (2D) MXene-based nanostructured materials: Synthesis strategies, properties, and applications as efficient pseudo-supercapacitors. *Chem. Eng. J.* **2023**, *472*, 144913. [CrossRef]
2. Wang, R.; Gao, J.; Vijayalakshmi, M.; Tang, H.; Chen, K.; Reddy, C.V.; Kakarla, R.R.; Anjana, P.; Rezakazemi, M.; Cheolho, B. Metal–organic frameworks and their composites: Design, synthesis, properties, and energy storage applications. *Chem. Eng. J.* **2024**, *496*, 154294. [CrossRef]
3. Lipu, M.H.; Hannan, M.A.; Hussain, A.; Hoque, M.; Ker, P.J.; Saad, M.H.M.; Ayob, A. A review of state of health and remaining useful life estimation methods for lithium-ion battery in electric vehicles: Challenges and recommendations. *J. Clean. Prod.* **2018**, *205*, 115–133. [CrossRef]
4. Liu, F.; Shao, C.; Su, W.; Liu, Y. Online joint estimator of key states for battery based on a new equivalent circuit model. *J. Energy Storage* **2022**, *52*, 104780. [CrossRef]
5. Liu, K.; Gao, Y.; Zhu, C.; Li, K.; Fei, M.; Peng, C.; Zhang, X.; Han, Q.-L. Electrochemical modeling and parameterization towards control-oriented management of lithium-ion batteries. *Control Eng. Pract.* **2022**, *124*, 105176. [CrossRef]
6. Zheng, Y.; Cui, Y.; Han, X.; Ouyang, M. A capacity prediction framework for lithium-ion batteries using fusion prediction of empirical model and data-driven method. *Energy* **2021**, *237*, 121556. [CrossRef]

7. Grimaldi, A.; Minuto, F.D.; Perol, A.; Casagrande, S.; Lanzini, A. Ageing and energy performance analysis of a utility-scale lithium-ion battery for power grid applications through a data-driven empirical modelling approach. *J. Energy Storage* **2023**, *65*, 107232. [CrossRef]
8. Jin, X.; Vora, A.; Hoshing, V.; Saha, T.; Shaver, G.; García, R.E.; Wasynczuk, O.; Varigonda, S. Physically-based reduced-order capacity loss model for graphite anodes in Li-ion battery cells. *J. Power Sources* **2017**, *342*, 750–761. [CrossRef]
9. Reniers, J.M.; Mulder, G.; Howey, D.A. Review and performance comparison of mechanical-chemical degradation models for lithium-ion batteries. *J. Electrochem. Soc.* **2019**, *166*, A3189–A3200. [CrossRef]
10. Wen, P.; Ye, Z.-S.; Li, Y.; Chen, S.; Xie, P.; Zhao, S. Physics-informed neural networks for prognostics and health management of lithium-ion batteries. *IEEE Trans. Intell. Veh.* **2023**, *9*, 2276–2289. [CrossRef]
11. Li, Y.; Liu, K.; Foley, A.M.; Zülke, A.; Berecibar, M.; Nanini-Maury, E.; Van Mierlo, J.; Hoster, H.E. Data-driven health estimation and lifetime prediction of lithium-ion batteries: A review. *Renew. Sustain. Energy Rev.* **2019**, *113*, 109254. [CrossRef]
12. 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]
13. Feng, J.; Cai, F.; Li, H.; Huang, K.; Yin, H. A data-driven prediction model for the remaining useful life prediction of lithium-ion batteries. *Process Saf. Environ. Prot.* **2023**, *180*, 601–615. [CrossRef]
14. How, D.N.; Hannan, M.A.; Lipu, M.S.H.; Sahari, K.S.; Ker, P.J.; Muttaqi, K.M. State-of-charge estimation of li-ion battery in electric vehicles: A deep neural network approach. *IEEE Trans. Ind. Appl.* **2020**, *56*, 5565–5574. [CrossRef]
15. Feng, X.; Weng, C.; He, X.; Han, X.; Lu, L.; Ren, D.; Ouyang, M. Online state-of-health estimation for Li-ion battery using partial charging segment based on support vector machine. *IEEE Trans. Veh. Technol.* **2019**, *68*, 8583–8592. [CrossRef]
16. 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]
17. Zhang, Y.; Feng, X.; Zhao, M.; Xiong, R. In-situ battery life prognostics amid mixed operation conditions using physics-driven machine learning. *J. Power Sources* **2023**, *577*, 233246. [CrossRef]
18. Hong, J.; Lee, D.; Jeong, E.-R.; Yi, Y. Towards the swift prediction of the remaining useful life of lithium-ion batteries with end-to-end deep learning. *Appl. Energy* **2020**, *278*, 115646. [CrossRef]
19. Zhao, W.; Ding, W.; Zhang, S.; Zhang, Z. A deep learning approach incorporating attention mechanism and transfer learning for lithium-ion battery lifespan prediction. *J. Energy Storage* **2024**, *75*, 109647. [CrossRef]
20. Lin, M.; You, Y.; Meng, J.; Wang, W.; Wu, J.; Stroe, D.-I. Lithium-ion battery degradation trajectory early prediction with synthetic dataset and deep learning. *J. Energy Chem.* **2023**, *85*, 534–546. [CrossRef]
21. Severson, K.A.; Attia, P.M.; Jin, N.; Perkins, N.; Jiang, B.; Yang, Z.; Chen, M.H.; Aykol, M.; Herring, P.K.; Fraggedakis, D. Data-driven prediction of battery cycle life before capacity degradation. *Nat. Energy* **2019**, *4*, 383–391. [CrossRef]
22. Hasib, S.A.; Islam, S.; Chakrabortty, R.K.; Ryan, M.J.; Saha, D.K.; Ahamed, M.H.; Moyeen, S.I.; Das, S.K.; Ali, M.F.; Islam, M.R. A comprehensive review of available battery datasets, RUL prediction approaches, and advanced battery management. *IEEE Access* **2021**, *9*, 86166–86193. [CrossRef]
23. Ma, G.; Zhang, Y.; Cheng, C.; Zhou, B.; Hu, P.; Yuan, Y. Remaining useful life prediction of lithium-ion batteries based on false nearest neighbors and a hybrid neural network. *Appl. Energy* **2019**, *253*, 113626. [CrossRef]
24. Su, L.; Wu, M.; Li, Z.; Zhang, J. Cycle life prediction of lithium-ion batteries based on data-driven methods. *ETransportation* **2021**, *10*, 100137. [CrossRef]
25. Ketkar, N.; Moolayil, J.; Ketkar, N.; Moolayil, J. Convolutional neural networks. In *Deep Learning With Python: Learn Best Practices of Deep Learning Models with PyTorch*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 197–242.
26. O'Shea, K. An introduction to convolutional neural networks. *arXiv* **2015**, arXiv:1511.08458.
27. Simonyan, K. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
28. Torrey, L.; Shavlik, J. Transfer learning. In *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*; IGI Global: Hershey, PA, USA, 2010; pp. 242–264. [CrossRef]
29. Ji, S.; Tan, K.Y.; Inaba, S.; Smith, H. In Performance Comparison of Commercially-Available Rechargeable 2032 Batteries, APS March Meeting Abstracts, 2023; p G00. 079.
30. Ma, S.; Jiang, M.; Tao, P.; Song, C.; Wu, J.; Wang, J.; Deng, T.; Shang, W. Temperature effect and thermal impact in lithium-ion batteries: A review. *Prog. Nat. Sci. Mater. Int.* **2018**, *28*, 653–666. [CrossRef]
31. Li, Z.; Liu, F.; Yang, W.; Peng, S.; Zhou, J. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 6999–7019. [CrossRef] [PubMed]
32. Weiss, K.; Khoshgoftaar, T.M.; Wang, D. A survey of transfer learning. *J. Big Data* **2016**, *3*, 9. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.