

HybridoNet-Adapt: A Domain-Adapted Framework for Accurate Lithium-Ion Battery RUL Prediction

Khoa Tran^{a,*}, Bao Huynh^a, Tri Le^a, Lam Pham^b, Vy-Rin Nguyen^c

^a*AIWARE Limited Company, 17 Huynh Man Dat Street, Hoa Cuong Bac Ward, Hai Chau District, Danang 550000, Vietnam*

^b*AIT Austrian Institute of Technology GmbH, Giefinggasse 4, Vienna 1210, Austria*

^c*Software Engineering Department - FPT University, Da Nang, 550000, Vietnam*

Abstract

Accurate prediction of the remaining useful life (RUL) in Lithium-ion battery (LIB) health management systems is crucial for ensuring reliability and safety. Current methods typically assume that training and testing data share the same distribution, overlooking the benefits of incorporating diverse data sources to enhance model performance. To address this limitation, we introduce a data-independent RUL prediction framework along with its domain adaptation (DA) approach, which leverages heterogeneous data sources for improved target predictions. Our approach integrates comprehensive data preprocessing—comprising feature extraction, denoising, and normalization—with a data-independent prediction model that combines Long Short-Term Memory (LSTM), Multihead Attention, and a Neural Ordinary Differential Equation (NODE) block, termed *HybridoNet*. The domain-adapted version, *HybridoNet-Adapt*, is trained using a novel technique inspired by the Domain-Adversarial Neural Network (DANN) framework, a regression ensemble method, and Maximum Mean Discrepancy (MMD) to learn domain-invariant features from labeled cycling data in the source and target domain. Experimental results demonstrate that our approach outperforms state-of-the-art techniques, providing reliable RUL predictions for real-world applications.

Keywords—Battery Health Management, Lithium-ion Batteries, Remaining Useful Life, Signal Processing, Data-Independent Prediction

*Corresponding author: khoa.tran@aiware.website

1. Introduction

1.1. Motivations

Lithium-ion batteries (LIBs) [11], renowned for their affordability and high energy density, are extensively utilized [14, 8, 50, 23] in electric vehicles (EVs), mobile phones, and energy storage stations. The global lithium-ion battery (LIB) market is projected to surpass 170 billion dollars by 2030 [33]. With the wide-ranging adoption of LIBs, interest in battery health management (BHM) has surged within both academia and industry in recent years. In a BHM system, several common and essential techniques are employed, including thermal management [21, 49], fault diagnosis/detection [7], state of charge (SOC) and state of health (SOH) estimation [28], RUL prediction [24, 31], and cycle life early prediction [38, 16, 29]. In the lifespan of a LIB, RUL prediction aims to forecast the remaining life, which can be measured by capacity or the number of remaining cycles the battery cell can undergo before reaching the end of life (EOL). RUL prediction methods fall into three categories: model-based, data-driven, and hybrid approaches.

Traditional model-based approaches often utilize physics-based degradation models, such as the Double Exponential Model (DEM) [29], two-phase degradation models [42], and Markov Models [51], constructed using early-cycle data (200-500 cycles). These models aim to forecast the entire battery's capacity degradation curve. However, relying solely on maximum discharge capacity degradation and early-cycle data often leads to inaccuracies due to the influence of various factors (current, voltage, temperature, time) and sudden changes in degradation trends [29, 42].

Data-driven models predict the RUL of LIBs by analyzing present cycling data. Techniques like dual-input Deep Neural Networks (DNN) [46], 1D Convolutional Neural Networks (1DCNN) [22], Dense layers [19], Long Short-Term Memory (LSTM) networks [13], and Echo State Networks (ESN) [18] have shown superior performance. These methods require large amounts of high-quality data.

Hybrid approaches combine model-based and data-driven methods to improve RUL prediction. For instance, a hybrid model using the Double Exponential Degradation Model (DEDM) and Gated Recurrent Unit (GRU) network fused with a Bayesian neural network (BNN) offers enhanced predictions [25]. Despite their advantages, hybrid models still depend on early-cycle data, limiting their flexibility.

Accurately predicting the RUL of LIBs remains challenging due to their nonlinear degradation processes and the need for highly accurate prediction models. The specific challenges will be discussed in the next section.

1.2. Problem Statement

As summarized in Table 1, current state-of-the-art studies categorize RUL prediction methods into two primary approaches: historical data-independent methods, which estimate the current RUL based on present cycling data and a few preceding cycles, and historical data-dependent methods, which leverage extensive early-cycle data to predict the battery’s full lifespan.

While historical data-dependent methods achieve reasonable accuracy in benchmark evaluations [25, 43], they are hindered by practical issues such as incomplete cycle data [31], unavailability of early-cycle records, vary operational conditions [16] throughout a battery’s lifespan, and challenges in battery repurposing [37]. Therefore, historical data-independent approaches are more suitable for real-world scenarios.

Table 1: Overview of RUL Prediction Methods in LIBs Research

Task					
Method	Prediction targets	Reference	Dataset	Signal Preprocessing	Prediction Model
Historical data-independent approach	RUL (capacity)	[1]	NASA PCoE [32] Toyota Research Institute (TRI) [38]	Feature extraction of temperature, current and voltage Sliding window technique for denoising	Cascaded forward neural network (CFNN)
	RUL (remaining cycles)	[46]	2022 Li-Ion Health Prediction (LHP) [30]	Feature-based condition extraction Sequential Feature Sampling	Dual-input DNN
	RUL (remaining cycles)	[31]	Oxford Battery [53]	Ageing-correlated parameter extraction	Physics-based DNN
Historical data-dependent approach	RUL (capacity)	[34]	2016 NASA Battery [52]	CEEMDAN	Single-input PA-LSTM
	RUL (capacity)	[42]	2007 NASA Battery [36]	Binary segmentation using particle filtering method	Two-phase capacity degradation model
	RUL (capacity)	[26]	CALCE-CX2 and CALCE-CS2 [40]	EMD GRU-FC	CNN model predicting the maximum discharging capacity
	RUL (capacity)	[43]	NASA PCoE [32] CALCE-CX2 and CALCE-CS2 [40]	Variational Modal Decomposition (VMD) Kullback-Leibler (KL) divergence	Bayesian optimization LSTM network
	RUL (capacity)	[25]	NASA PCoE [32] NASA Random Walk [2]	Z-score normalization EMD	ESN GRU-CNN network Double Exponential Degradation Model (DEDM) BNN
	Cycle life	[38]	TRI [38]	Feature extraction for the first 100 cycles	Elastic net
	Cycle life	[16]	TRI [38]	Statistical and gradient-based feature extraction	Extreme Gradient Boosting (XGBoost)
	Cycle life, RUL (capacity)	[29]	TRI [38]	Spline function (interpolates discharge capacities to length 1000)	CNN model Double exponential model (DEM) Gaussian process regression (GPR)

Many existing methods focus on predicting RUL based on estimated maximum discharge capacity thresholds. However, accurately forecasting the remaining cycles provides clearer insights into the battery’s usable life.

Datasets such as the TRI dataset (124 cells, fast-charging) and the LHP dataset (77 cells, diverse discharge) provide extensive charge-discharge scenarios, making them well-suited for both training and validation of data-driven models. In contrast, smaller datasets like the Oxford Battery dataset (13 cells) and NASA battery datasets (4–34 cells) limit model robustness in real-world failure prediction.

Signal preprocessing techniques are generally categorized into statistical feature extraction methods, such as mean and standard deviation (Std), and variational decomposition methods like Empirical Mode Decomposition (EMD) and Variational Mode Decomposition (VMD). While variational decomposition increases the complexity of the input signal and often maintains or enlarges its original size, statistical feature extraction significantly reduces the signal’s dimensionality and effectively captures key signal characteristics, making it well-suited for real-time applications and high-accuracy predictions in industrial settings.

Model-based and hybrid approaches typically rely on early-cycle data for RUL prediction, yet each battery exhibits unique degradation patterns over its lifespan requiring adaptive data-driven strategies. Moreover, Transfer learning methods such as domain-adversarial neural networks (DANN) [10, 47] and Generative Adversarial Networks (GANs) [12] offer effective solutions for adapting to other source’s degradation patterns to improve RUL prediction in the target domain. To address these challenges, our proposed approach will be represented in the next section.

1.3. Main Contribution

We propose a historical data-independent method that leverages only present and recent cycling data, eliminating the need for early-cycle data while ensuring accurate RUL forecasts. Additionally, we conduct large-scale experiments on two largest available datasets of A123 APR18650M1A cells [38, 46], covering diverse charging and discharging conditions. These evaluations validate the effectiveness and real-world applicability of our data-driven approach. Our key contributions include:

1. **A RUL Prediction Module:** We propose a comprehensive RUL prediction framework that consists of a signal processing phase and a regression neural network, named HybridoNet. The signal processing phase extracts key degradation features from raw battery signals, while HybridoNet, composed of LSTM, Multihead Attention [41], and Neural Ordinary Differential Equation (NODE) [4] blocks. This data-driven approach achieves superior accuracy compared to existing methods.
2. **Enhancing Neural Networks with Domain Adaptation:** To further improve generalization across different battery degradation conditions, we introduce HybridoNet-Adapt, an extension of HybridoNet incorporating a Domain Adaptation (DA) technique inspired by DANN

[10]. This adaptation mechanism helps the model learn transferable representations, mitigating discrepancies between different operating conditions and improving RUL prediction in real-world applications.

2. Preliminaries

This section provides an overview of HybridoNet’s key components, including LSTM, Multihead Attention, and NODE. It also represents the DA-based strategy that inspires HybridoNet-Adapt.

2.1. LSTM:

A recurrent neural network (RNN) [35] architecture designed to overcome the vanishing gradient problem by introducing gating mechanisms. Its operations are defined by:

$$\begin{aligned} i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i), \\ f_t &= \sigma(W_f x_t + U_f h_{t-1} + b_f), \\ o_t &= \sigma(W_o x_t + U_o h_{t-1} + b_o), \\ \tilde{c}_t &= \tanh(W_c x_t + U_c h_{t-1} + b_c), \\ c_t &= f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \\ h_t &= o_t \odot \tanh(c_t), \end{aligned}$$

where x_t is the input at time t , h_{t-1} is the previous hidden state, and σ and \tanh are the sigmoid and hyperbolic tangent activation functions, respectively.

2.2. Multihead Attention:

A critical mechanism in Transformer models, enabling the network to attend jointly to information from different subspaces. The basic building block is the scaled dot-product attention:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V,$$

where Q , K , and V represent the query, key, and value matrices, respectively, and d_k is the dimensionality of the keys. In a multihead setting, multiple attention heads are computed:

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V),$$

and their outputs are concatenated and linearly transformed:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O.$$

2.3. NODE:

A framework that extends deep learning architectures to model continuous dynamics. In NODE, the evolution of a hidden state $h(t)$ is governed by an ordinary differential equation:

$$\frac{dh(t)}{dt} = f(h(t), t, \theta),$$

where f is a neural network parameterized by θ . The final state is obtained by solving this ODE over a time interval, which provides a flexible and memory-efficient representation.

2.4. DANN:

A framework facilitates domain adaptation by learning domain-invariant features. It encourages the feature extractor G_F to generate features that are indistinguishable across domains. The overall objective is formulated as:

$$\begin{aligned} \min_{G_F, G_Y} \max_{G_D} \mathcal{L}(G_F, G_Y, G_D) = & \mathbb{E}_{(x, y) \sim S} [\mathcal{L}_y(G_Y(G_F(x)), y)] \\ & - \lambda \mathbb{E}_{x \sim S \cup T} [\mathcal{L}_d(G_D(G_F(x)), d)], \end{aligned}$$

where G_Y is the label predictor, G_D is the domain discriminator, S and T denote the source and target domains, d represents domain labels, and λ balances classification and domain confusion losses.

3. Proposed method

3.1. Overall architecture

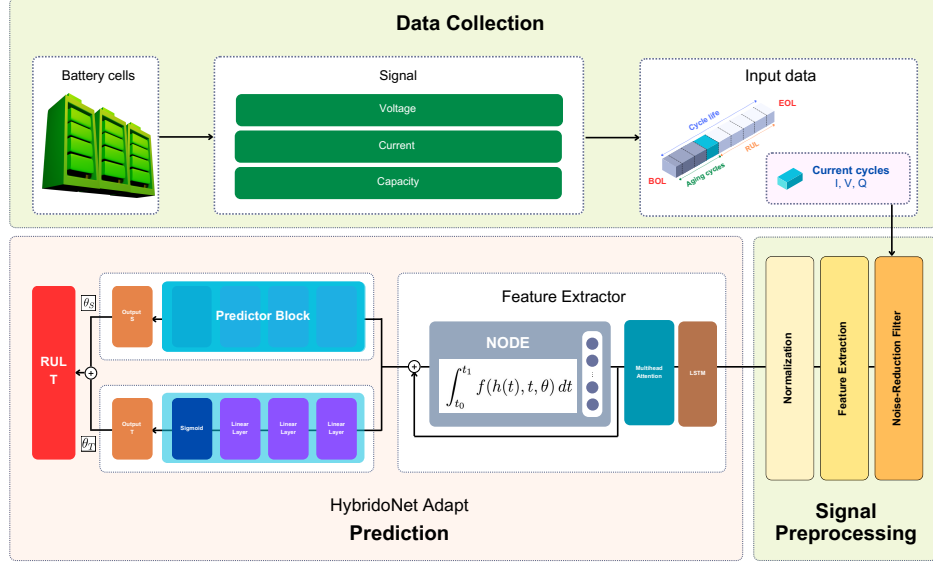


Figure 1: Overall architecture of our RUL prediction process for Lithium-ion battery cells.

Figure 1 illustrates the RUL prediction process for Lithium-ion battery cells. In the data collection phase, Lithium Iron Phosphate (LFP)/graphite cells are monitored, capturing voltage, current, and capacity signals of each cycle. The cycle life of a battery is defined as the number of cycles from the Beginning of Life (BOL) to the EOL, typically when the maximum capacity degrades to 70% [34] or 80% [29] of its nominal capacity. The RUL is computed as: $RUL = \text{Cycle life} - \text{Aging cycles}$.

The signal preprocessing phase begins with applying a median filter [20] to smooth out sudden peaks in the raw signal vectors. Each signal vector—voltage, current, and capacity—is then processed through feature extraction methods, including mean, standard deviation (Std), minimum (Min), maximum (Max), variance (Var), and median (Med) [9, 39]. The representation of data for each cycle is given by $X^i = [x_{\text{current}}^i, x_{\text{voltage}}^i, x_{\text{capacity}}^i]$, $X^i \in \mathbb{R}^{3 \times 6}$, where 3 corresponds to the three signal types (voltage, current, and capacity), and 6 represents the extracted features (Mean, Std, Min, Max,

Var, Med). Each input sample for the RUL prediction model consists of 10 selected cycles sampled from a 30-cycle window (one cycle every three cycles) [30]. The input sample can be represented as $\mathbf{X}^i = [X^1, X^2, \dots, X^{10}]$, $\mathbf{X} \in \mathbb{R}^{10 \times 3 \times 6}$. The output's shape of input data after feature extraction step is $\mathbf{X} \in \mathbb{R}^{N \times 10 \times 3 \times 6}$, where N denotes the number of samples. In the normalization step, MinMaxScaler is applied. \mathbf{X} is first reshaped from $\mathbb{R}^{N \times 10 \times 3 \times 6}$ to $\mathbb{R}^{(N \times 10) \times 18}$, where 18 represents the flattened 6×3 feature matrix. MinMaxScaler is then fitted and applied to scale the data between 0 and 1. Finally, the normalized data is reshaped back to $\mathbb{R}^{N \times 10 \times 3 \times 6}$ for further processing.

In the prediction phase, the RUL prediction model, HybridoNet-Adapt, consisting of multiple blocks, is used to map \mathbf{X} to the predicted RUL $\mathbf{Y} \in \mathbb{R}^{N \times 1}$. The details of the proposed RUL prediction models are presented in the following sections.

3.2. HybridoNet: A Proposed RUL model

HybridoNet consists of two key components: a predictor G_Y^T and a feature extractor G_F module. The feature extractor integrates an LSTM, a Multihead Attention mechanism, and a NODE block. In the NODE block, a linear layer is employed to approximate the functionality of multiple linear layers, thereby reducing computational overhead. This block applies a continuous transformation function: $\frac{dh(t)}{dt} = f(h(t), t, \theta)$, where t represents the time steps, θ denotes the trainable parameters of f , and f is a trainable function. In our NODE block, f is implemented as a linear layer. The initial condition of the NODE block is given by $h(t_0)$. Solving the ODE over the interval $[t_0, t_1]$ produces the transformed state $h(t_1)$, which serves as the NODE block output:

$$h(t_1) = h(t_0) + \int_{t_0}^{t_1} f(h(t), t, \theta) dt.$$

Where, t_0 and t_1 are set to 0 and 1, respectively, based on experimental findings (see Figure 7).

The predictor module consists of three linear layers, each followed by ReLU [5] activation, 1D Batch Normalization [17], and Dropout [15], respectively, with a sigmoid activation [48] layer at the end. The HybridoNet model is trained using the Mean Squared Error (MSE) [44] loss function to optimize the model via an optimizer. The MSE loss is defined as:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N} \sum_{i=1}^N (G_Y^S(G_F(X_i)) - Y_i)^2$$

where Y_i is the RUL ground truth. Note that the proposed HybridoNet is not the main focus of this paper. Instead, we primarily introduce HybridoNet-Adapt, a DA-based extension of HybridoNet, which will be presented in the next section.

3.3. HybridoNet-Adapt: An Enhanced Version Leveraging Domain Adaptation

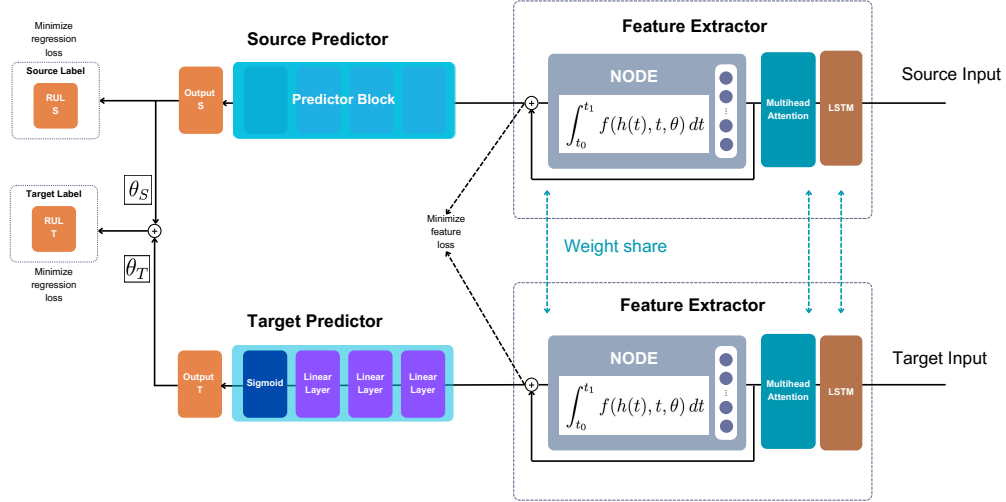


Figure 2: The training process for the proposed HybridoNet-Adapt.

As shown in Figure 2, HybridoNet-Adapt extends HybridoNet with a DA technique. It includes a source predictor G_Y^S , a target predictor G_Y^T , and a feature extractor G_F , all sharing the same architecture as their counterparts in HybridoNet (Section 3.2). The target and source prediction in HybridoNet-Adapt are computed as:

$$\hat{Y}_i^T(X_i) = \theta^S G_Y^S(G_F(X_i)) + \theta^T G_Y^T(G_F(X_i)), \quad (1)$$

$$\hat{Y}_i^S(X_i) = G_Y^S(G_F(X_i)). \quad (2)$$

Here, θ^S and θ^T are trainable trade-off parameters that balance the contributions of the source and target predictors, and \hat{Y}_i represents the predicted output.

The HybridoNet-Adapt is optimized by two losses: the mean squared error (MSE) loss, \mathcal{L}_{MSE} , for the regression outputs, and the maximum mean discrepancy (MMD) loss, \mathcal{L}_{MMD} , to align feature embeddings. The total loss is defined as follows:

$$\begin{aligned}\mathcal{L}(X_i^S, X_i^T, Y_i^S, Y_i^T) = & \mathcal{L}_{\text{MSE}}(\hat{Y}_i^S(X_i^S), Y_i^S) \\ & + \mathcal{L}_{\text{MSE}}(\hat{Y}_i^T(X_i^T), Y_i^T) \\ & + \lambda \mathcal{L}_{\text{MMD}}(G_F(X_i^S), G_F(X_i^T)).\end{aligned}$$

Here, X_i^S and X_i^T represent input samples from the source and target domains, respectively, and Y_i^S and Y_i^T denote the corresponding RUL ground truth values.

The MMD loss measures the discrepancy between the distributions of feature embeddings. Given feature sets $\{X_i^S\}_{i=1}^n$ from the source domain and $\{X_i^T\}_{i=1}^m$ from the target domain, it is defined as:

$$\mathcal{L}_{\text{MMD}} = \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n k(X_i^S, X_j^S) + \frac{1}{m^2} \sum_{i=1}^m \sum_{j=1}^m k(X_i^T, X_j^T) - \frac{2}{nm} \sum_{i=1}^n \sum_{j=1}^m k(X_i^S, X_j^T),$$

where $k(\cdot, \cdot)$ is a kernel function, commonly chosen as the Gaussian kernel: $k(x, y) = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)$, with σ as the kernel bandwidth parameter.

The hidden dimension of the LSTM and Multihead Attention is set to 128, with 2 LSTM layers. The hidden dimensions of the linear layers in the predictor are [128, 64, 32, 1], with a dropout rate of 0.1. These values are determined based on the experimental results presented in the next section.

In the next section, multiple experiments are conducted to determine the optimal configuration of HybridoNet-Adapt, and demonstrating the superiority over state-of-the-art methods.

4. Experiments and discussion

4.1. Experimental Setup

Our proposed RUL models are implemented using the PyTorch framework and optimized with the AdamW algorithm [27] to minimize the respective

loss functions. Experiments are performed on an NVIDIA A100 GPU with 80GB of memory. Each experiment runs for 10 epochs with a batch size of 128 and a fixed learning rate of 0.0005. To obtain a representative prediction, each experiment is repeated 10 times, and the average of their predictions is taken. The training dataset is split into 90% for training and 10% for validation, with the best selected based on the lowest RMSE (see 4.3) on the validation set. The weighting factor λ in 3.3 is dynamically adjusted during training using the following function: $\lambda = \frac{2}{1+e^{-10 \cdot \frac{\text{epoch}}{\text{epochs}}}} - 1$. Here, epoch is the current training iteration.

4.2. Datasets

4.2.1. First dataset: Varied fast-charging conditions, with consistent discharging conditions

The first dataset, referred to as the TRI dataset [38], encompasses a detailed study of 124 LFP/graphite lithium-ion batteries. Each LIB in the dataset has a nominal capacity of 1.1 Ah and a nominal voltage of 3.3 V. The cycle life span of these batteries ranges from 150 to 2,300 cycles, showcasing a wide spectrum of longevity. The dataset is divided into three distinct parts: a training set with 41 LIBs, a primary test set with 43 LIBs, and a secondary test set comprising 40 LIBs. In terms of operational conditions, all LIBs were subjected to uniform discharge protocols. Specifically, they were discharged at a constant current rate of 4 C until the voltage dropped to 2 V, followed by a constant voltage discharge at 2 V until the current diminished to C/50. The LIBs were charged at rates between 3.6 C and 6 C, under a controlled temperature of 30°C within an environmental chamber. The dataset contains approximately 96,700 cycles, making it one of the largest datasets to consider various fast-charging protocols.

4.2.2. Second dataset: Varied discharge conditions, with consistent fast-charging conditions

The second dataset, referred as the LHP dataset [30], was developed through a battery degradation experiment involving 77 cells (LFP/graphite A123 APR18650M1A) with a nominal capacity of 1.1 Ah and a nominal voltage of 3.3 V. Each of the 77 cells was subjected to a unique multi-stage discharge protocol, while maintaining an identical fast-charging protocol for all cells. The experiment was conducted in two thermostatic chambers at a controlled temperature of 30°C. The dataset encompasses a total of 146,122

discharge cycles, making it one of the largest datasets to consider various discharge protocols. The cells exhibit a cycle life ranging from 1,100 to 2,700 cycles, with an average of 1,898 cycles and a standard deviation of 387 cycles. The discharge capacity as a function of cycle number reveals a wide distribution of cycle lives. For analytical purposes, the cells were divided into two groups: 55 cells were used for training, and 22 cells were reserved for testing. Both the observed cycle life and initial discharge capacity distributions approximate normal distributions with significant standard deviations.

4.3. Evaluation Metrics

To evaluate RUL prediction, we use Root Mean Square Error ($RMSE$) [45], R-squared (R^2) [3, 46, 43], and Mean Absolute Percentage Error ($MAPE$) [6]. These are calculated as follows:

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

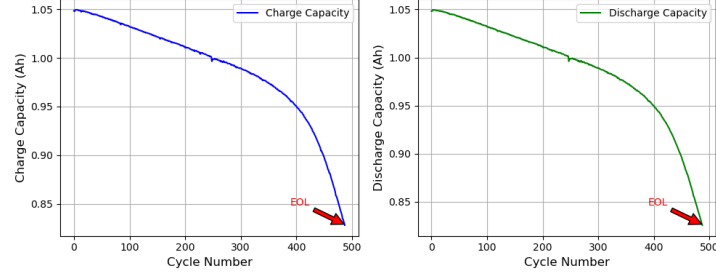
$$MAPE(y_i, \hat{y}_i) = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y} \times 100,$$

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

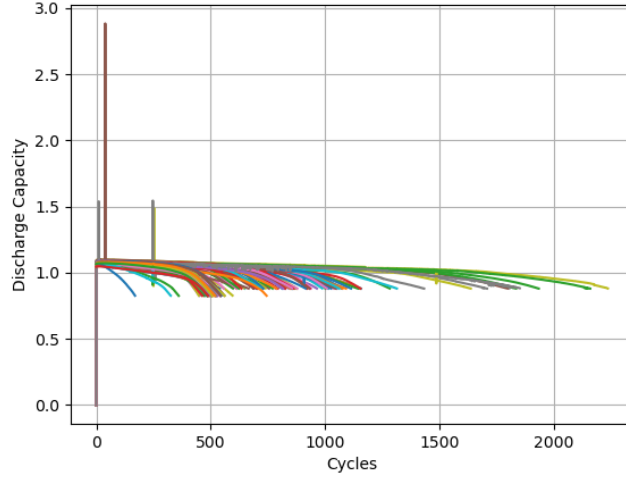
Where y_i and \hat{y}_i are the observed and predicted RUL, respectively. y is cycle life. The smaller the $RMSE$ and $MAPE$, and the larger the R^2 , the better the performance.

4.4. Data Analysis and Preprocessing

Figures 3a and 3b analyze battery cycle life. Figure 3a tracks an individual cell's charge and discharge capacities, marking EOL when the maximum capacity degrades to 80% of nominal capacity. Figure 3b compares cycle life across cells, revealing significant variation in discharge capacity. This variability challenges prediction models for RUL, emphasizing the need for accurate predictions for BHM systems.



(a) Maximum charge and discharge capacities over charge-discharge cycles for a single battery cell



(b) Maximum discharge capacities over charge-discharge cycles for many battery cells in the first dataset

Figure 3: Comparison of maximum charge and discharge capacities over cycle life for battery cells.

Before feature extraction, the raw signals exhibit sudden peaks and fluctuations, resembling noise. Smoothing the time-series data can help reduce noise and enhance the key characteristics of the signal. To achieve this, a median filtering method is applied to eliminate abrupt peaks in the signals before feature extraction. As a result, the application of median filtering improves overall model performance. The filtered data leads to better RMSE, R^2 , and MAPE (%) values compared to the unfiltered data, as illustrated in Figure 4.

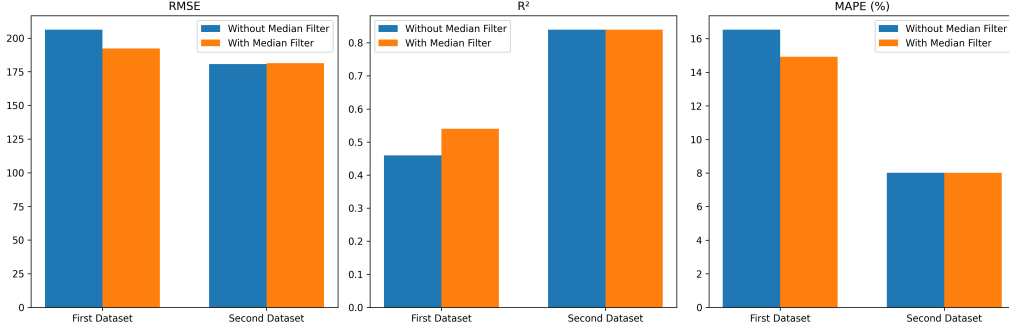


Figure 4: Comparison of RUL prediction performance with and without median filtering.

4.5. Feature Extractor

The feature extractor in both HybridoNet and HybridoNet-Adapt is progressively developed, starting with an LSTM architecture and sequentially integrating Multihead Attention (MA) and a NODE block. To evaluate the effectiveness of each component, we assess the performance of HybridoNet at different stages. With each addition, the model’s predictive capability improves. Ultimately, HybridoNet achieves an RMSE of 166.33, an R^2 score of 0.86, and a MAPE of 7.44%, demonstrating its superior performance.

Model	RMSE ↓	R^2 ↑	MAPE (%) ↓
LSTM	176,66	0,83	8,35
LSTM + MA	167,01	0,85	7,53
LSTM + MA + NODE	166.33	0.86	7.44

Table 2: Performance comparison of LSTM-based models on the testing data of the second dataset.

4.6. HybridoNet-Adapt: An Extension of HybridoNet for Domain Adaption

HybridoNet-Adapt is evaluated with various feature loss functions, including CORAL Loss, Domain Loss [10], MMD, as well as combinations such as MMD with Domain Loss and MMD with Domain Loss and CORAL Loss, as shown in Figure 6. The results indicate that using only MMD as

the feature loss function yields the best performance, achieving an RMSE of 160.05.

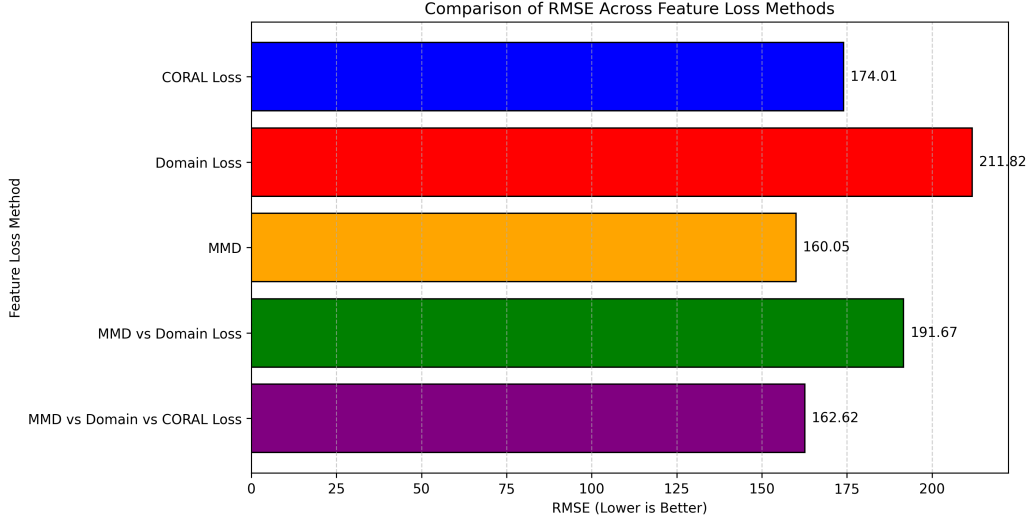


Figure 5: Comparison of different feature loss methods.

To determine the optimal hyperparameters, including hidden dimension of all layers, the number of recurrent LSTM layers, and the dropout rate, 27 experiments were conducted. The results are presented in Figure 6. In the graph, L represents the number of recurrent layers, H denotes the hidden dimension size. Based on RMSE score, the best performance is achieved with 2 recurrent LSTM layers, a hidden dimension of 64, and a dropout rate of 0.1.

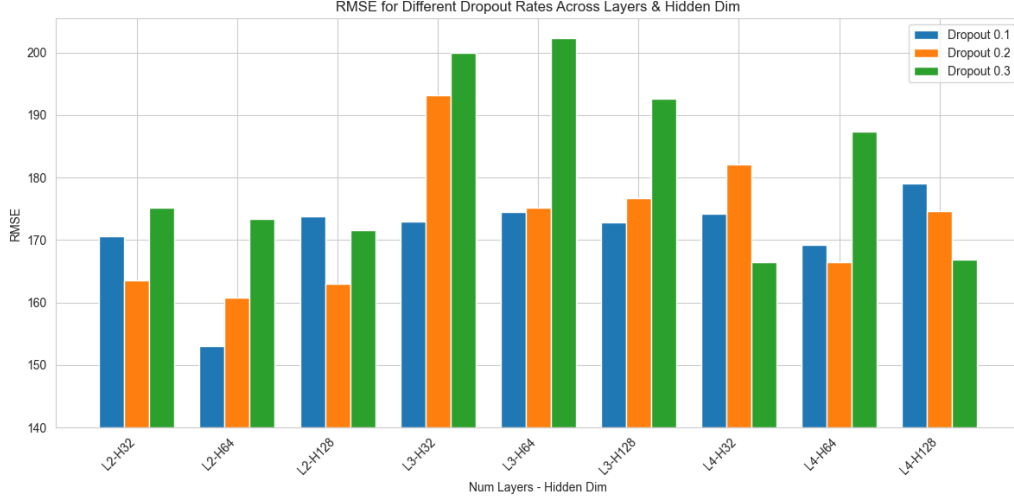


Figure 6: Comparison of RMSE across different number of LSTM layers, hidden dimensions, and dropout configurations.

To identify the optimal time step in the sequence dimension for both MA and NODE outputs, a comprehensive evaluation was conducted. Various NODE output time steps ranging from 2 to 6 were tested, along with different MA output time step selections, including the last, the second-to-last, and the mean time step. As shown in Figure 7, the best performance was achieved when using the second-to-last time step of the MA output and a NODE output time step of 2 ($t + 2$).

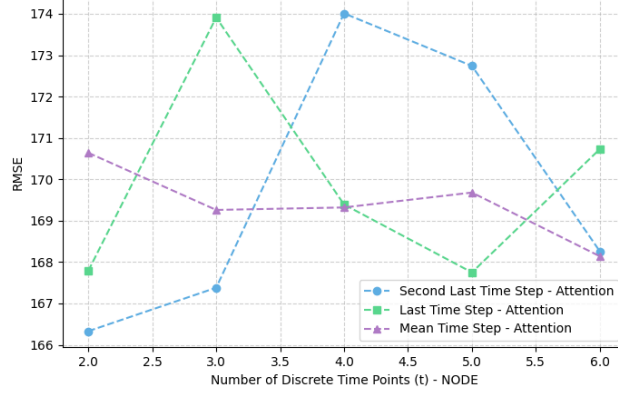


Figure 7: Comparison of RMSE for different NODE discrete time steps (t) and Multihead Attention output time step selections.

The proposed HybridoNet-Adapt model is systematically evaluated under various scenarios by experimenting with four different target datasets, each derived from the training data of the second dataset. The source data is the training data from the first dataset. Below are four groups of battery cells selected from the training data of the second dataset. These groups are carefully formed to ensure each set represents a diverse range of battery performance. For instance, Group 1 includes both high-cycle cells (e.g., 2-2 with 2,651 cycles) and low-cycle cells (e.g., 1-6 with 1,143 cycles), ensuring a comprehensive representation of aging behaviors.

- **Group 1:** 1-3 (1,858 cycles), 1-6 (1,143 cycles), 2-2 (2,651 cycles), 2-6 (1,572 cycles), 3-2 (2,283 cycles), 3-6 (2,491 cycles), 4-3 (1,142 cycles), 5-4 (1,962 cycles)
- **Group 2:** 1-5 (1,971 cycles), 1-8 (2,285 cycles), 2-4 (1,499 cycles), 2-7 (2,202 cycles), 3-3 (1,649 cycles), 3-7 (2,479 cycles), 4-4 (1,491 cycles), 5-5 (1,583 cycles)
- **Group 3:** 2-8 (1,481 cycles), 3-4 (1,766 cycles), 3-8 (2,342 cycles), 4-1 (2,217 cycles), 4-7 (2,216 cycles), 5-1 (2,507 cycles), 5-6 (2,460 cycles), 6-3 (1,804 cycles)
- **Group 4:** 4-8 (1,706 cycles), 5-2 (1,926 cycles), 5-7 (1,448 cycles), 6-4

(1,717 cycles), 6-5 (2,178 cycles), 7-2 (2,030 cycles), 7-7 (1,685 cycles), 8-2 (2,041 cycles)

- **All:** All battery cells from the training set of the second dataset.

Table 3 shows that HybridoNet-Adapt outperforms both HybridoNet (without DA) and DANN (with DA) across all groups. It achieves the lowest RMSE and MAPE while maintaining the highest R^2 , demonstrating better generalization. For instance, in Group 1, HybridoNet-Adapt reduces RMSE from 368.99 to 356.46 and improves R^2 from 0.21 to 0.30. On the full dataset, it achieves the best RMSE of 153.24 and R^2 of 0.88, significantly outperforming DANN, which shows degraded performance (RMSE = 835.35, $R^2 = -1.37$). DANN struggles with large variations in battery aging, while HybridoNet-Adapt effectively adapts to different distributions, leading to consistently better predictions.

Table 3: Comparison of HybridoNet, DANN, and HybridoNet-Adapt across four target data groups from the second dataset, based on testing scores from the testing data of the second dataset.

Group	HybridoNet (Without DA)			DANN (With DA)			HybridoNet-Adapt (With DA)		
	RMSE ↓	R^2 ↑	MAPE (%) ↓	RMSE ↓	R^2 ↑	MAPE (%) ↓	RMSE ↓	R^2 ↑	MAPE (%) ↓
Group 1	368.99	0.21	18.26	604.14	-0.36	27.26	356.46	0.30	17.35
Group 2	245.58	0.71	11.84	665.02	-0.51	29.33	240.90	0.73	11.09
Group 3	334.79	0.35	16.69	1007.93	-3.39	51.02	316.79	0.41	15.43
Group 4	304.91	0.61	14.08	758.16	-0.99	32.84	293.40	0.63	13.27
All	166.33	0.86	7.44	835.35	-1.37	36.99	153.24	0.88	7.30

Table 4 presents the evaluation metrics for RUL prediction on the test data from the second dataset, comparing Elastic Net, A_1 , A_2 of paper [30] (see Table S4), with our HybridoNet, and HybridoNet-Adapt methods. The results indicate that HybridoNet-Adapt achieves competitive RMSE values, particularly in cases where Elastic Net exhibits high errors. The R^2 values show that HybridoNet-Adapt generally improves predictive accuracy compared to the baseline methods. Additionally, MAPE results suggest that HybridoNet-Adapt provides more stable and reliable predictions, especially in challenging scenarios. Overall, these findings demonstrate the potential of HybridoNet-Adapt for enhanced RUL estimation.

Table 4: Evaluation metrics for RUL prediction performance on the test data from the second dataset, using existing Elastic Net, A_1 , and A_2 results of paper [30] (see Table S4).

Protocol	Channel	RMSE (cycles)					R ²					MAPE (%)				
		Elastic net	A ₁	A ₂	HybridoNet	HybridoNet-Adapt	Elastic net	A ₁	A ₂	HybridoNet	HybridoNet-Adapt	Elastic net	A ₁	A ₂	HybridoNet	HybridoNet-Adapt
#1	1-1	252	63.8	42.7	30.9	57.84	0.646	0.977	0.990	0.99	0.98	14.1	3.64	2.16	1.67	3.39
#2	1-2	722	262	272	483.29	514.39	0.102	0.882	0.873	0.6	0.55	23.2	8.64	8.52	15.22	17.34
#12	2-5	365	300	364	96.9	158.42	0.122	-	0.126	0.94	0.84	19.4	23.5	22.0	5.46	9.38
#16	3-1	422	104	133	327.53	129.66	0.407	0.964	0.941	0.65	0.94	17.0	4.43	6.00	15.39	6.42
#28	4-5	313	307	301	62.55	125.98	0.493	0.512	0.531	0.98	0.92	16.1	15.7	16.3	2.69	7.4
#34	5-3	737	279	301	346.43	392.7	0.0225	0.867	0.845	0.8	0.74	24.2	8.94	9.47	11.18	12.4
#39	6-1	310	147	120	70.95	34.57	0.532	0.896	0.930	0.98	0.99	14.3	7.69	6.01	3.36	1.69
#40	6-2	413	96.3	141	140.41	104.43	0.415	0.968	0.932	0.93	0.96	15.5	4.08	5.98	6.37	5.03
#44	6-6	672	400	349	226.53	248.97	0.0821	0.675	0.753	0.9	0.87	22.4	14.5	12.6	8.71	9.52
#45	6-8	609	334	359	148.63	217.09	0.236	0.769	0.734	0.95	0.9	20.3	12.6	13.6	5.23	8.3
#50	7-5	372	58.0	46.2	68.29	142.18	0.508	0.988	0.992	0.98	0.93	15.6	2.70	2.01	2.53	6.52
#51	7-6	372	295	263	50.22	81.06	0.129	0.453	0.566	0.98	0.96	21.5	15.7	15.6	2.65	5
#54	8-1	303	200	231	331.1	319.56	0.319	0.702	0.603	0.19	0.25	19.7	12.8	15.6	23.66	23.94
#58	8-5	281	45.3	57.1	229.23	227.22	0.449	0.986	0.977	0.64	0.64	17.7	2.64	3.30	15.19	16.5
#59	8-6	527	91.8	81.5	45.05	116.28	0.386	0.981	0.985	1	0.97	18.1	3.41	2.97	1.7	4.29
#61	8-8	412	363	382	254.98	49.44	0.245	0.411	0.349	0.71	0.99	18.2	18.2	19.3	13.3	2.31
#65	9-4	431	104	74.5	102.13	34.23	0.406	0.966	0.982	0.97	1	16.9	4.56	3.17	3.79	1.43
#67	9-6	403	297	292	182.32	56.22	0.328	0.634	0.649	0.86	0.99	18.4	15.4	15.1	6.24	2.97
#70	10-1	386	111	81.0	27.28	57.33	0.355	0.947	0.972	1	0.99	17.6	5.03	4.24	1.28	2.52
#73	10-4	331	67.5	60.9	184.35	43.13	0.582	0.983	0.986	0.87	0.99	14.3	3.15	2.95	8.36	2.05
#75	10-6	485	112	84.0	188.2	103.52	0.442	0.970	0.983	0.92	0.97	16.0	3.93	3.12	6.62	3.96
#76	10-7	405	86.7	52.8	62.06	157.16	0.353	0.970	0.989	0.98	0.9	16.7	3.25	1.97	3.13	8.28
Mean	-	434	192	186	166.33	153.24	0.344	0.795	0.804	0.86	0.88	18.1	8.84	8.72	7.44	7.3

Table 5 presents the performance comparison of different models on the secondary test data from the first dataset. HybridoNet-Adapt achieves the best results, with the lowest RMSE (146.52), highest R^2 score (0.72), and lowest MAPE (11.85%), demonstrating its superior predictive accuracy. HybridoNet outperforms XGBoost in all metrics, highlighting the effectiveness of deep learning-based approaches. The improvements in HybridoNet-Adapt further validate the benefits of domain adaptation in enhancing RUL prediction performance.

Model	RMSE ↓	R^2 ↑	MAPE (%) ↓
XGBoost	192.28	0.54	14.92
HybridoNet	153.47	0.68	12.71
HybridoNet-Adapt	146.52	0.72	11.85

Table 5: Evaluation metrics for RUL prediction performance on the secondary test data from the first dataset.

Figure 8 illustrates the RUL predictions of XGBoost, HybridoNet, HybridoNet-Adapt, and DANN, compared to the true (observed) RUL for Cell 4-5 and Cell 3-1 in the testing set of the second dataset. Among all methods, HybridoNet-Adapt demonstrates the closest alignment with the observed RUL, highlighting its superior predictive accuracy. This improvement is attributed to HybridoNet-Adapt’s ability to align feature representations from the source domain to the target domain, as shown in Figure 9. By effectively

increasing the amount of target-relevant data through our DA adaptation technique, HybridoNet-Adapt enhances robustness, making it more adaptable to diverse real-world battery degradation scenarios.

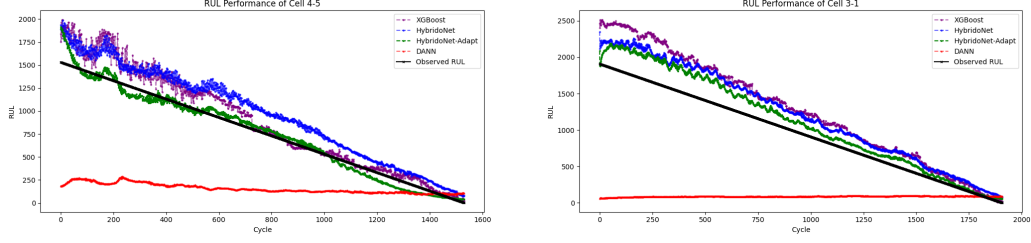


Figure 8: Comparison of model predictions with observed RUL for Cells 4-5 and 3-1 from the testing data of the second dataset.

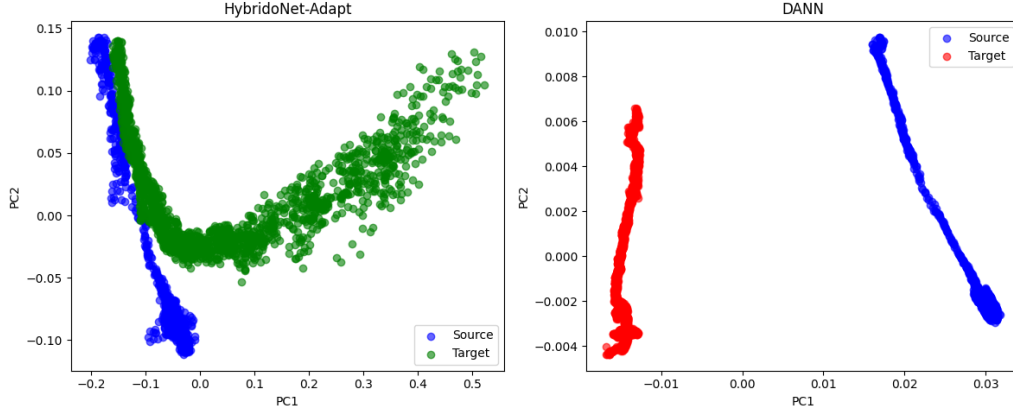


Figure 9: PCA-based comparison of embedding features between HybridoNet-Adapt and DANN.

4.7. Comparison with state-of-the-art methods

Figure 10 presents a comparison of our HybridoNet and HybridoNet-Adapt models with state-of-the-art methods, including Elastic Net [30], A_1 [30], A_2 [30], Ridge Linear [46], Random Forest [46], and Dual-input DNN [46]. The results demonstrate that our HybridoNet-Adapt achieves the lowest RMSE of 153.24, outperforming all other approaches, particularly the Dual-input DNN [46], which attains an RMSE of 159.84. This highlights the effectiveness of our proposed method in enhancing predictive performance.

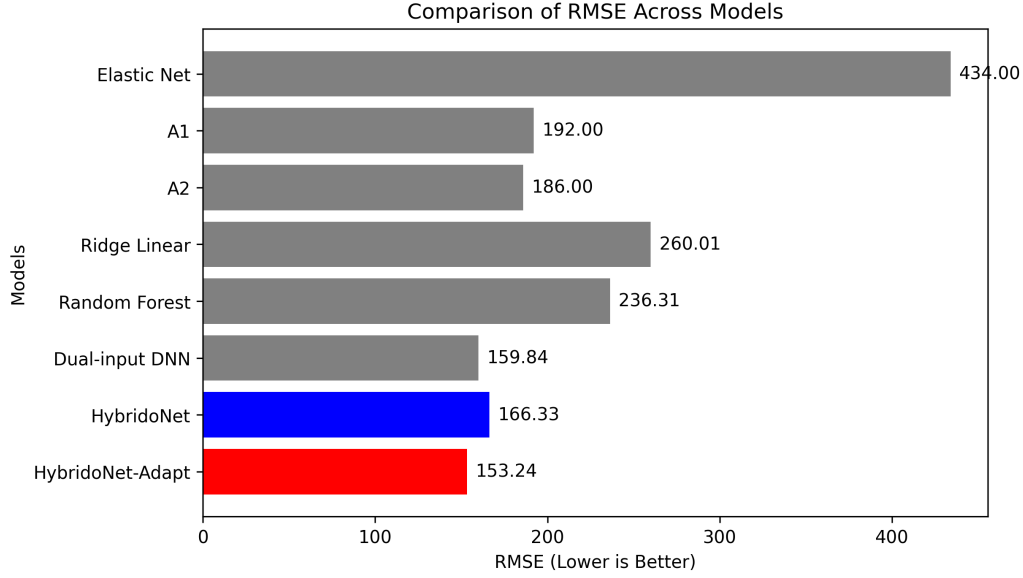


Figure 10: Comparison of our proposed models with existing state-of-the-art methods on the testing data of the second dataset.

5. Conclusion

In this study, we introduced HybridoNet, a novel deep learning framework that seamlessly integrates signal processing with LSTM, Multihead Attention, and NODE blocks to enhance RUL prediction accuracy. Additionally, we proposed HybridoNet-Adapt, an advanced variant incorporating domain adaptation inspired by DANN, significantly improving generalization across diverse battery degradation patterns. Our experimental results demonstrate the superiority of our approach over traditional machine learning models such as XGBoost and Elastic Net, as well as state-of-the-art deep learning methods like Dual-input DNN, highlighting its effectiveness in predictive maintenance. For future work, we plan to further enhance model generalization through self-supervised learning, refine real-time deployment strategies, and explore multi-modal data integration to improve scalability and robustness across various industrial applications.

Acknowledgment

Contact Information

For access to the code and further information about these proposed systems, please contact AIWARE Limited Company at: <https://aiware.website/Contact>

References

- [1] Shaheer Ansari, Afida Ayob, MS Hossain Lipu, Aini Hussain, Maher GM Abdolrasol, Muhammad Ammirul Atiqi Mohd Zainuri, and Mohamad Hanif Md Saad. Optimized data-driven approach for remaining useful life prediction of lithium-ion batteries based on sliding window and systematic sampling. *Journal of Energy Storage*, 73:109198, 2023.
- [2] Brian Bole, Chetan S Kulkarni, and Matthew Daigle. Adaptation of an electrochemistry-based li-ion battery model to account for deterioration observed under randomized use. In *Annual conference of the PHM society*, volume 6, 2014.
- [3] A Colin Cameron and Frank AG Windmeijer. An r-squared measure of goodness of fit for some common nonlinear regression models. *Journal of econometrics*, 77(2):329–342, 1997.
- [4] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.
- [5] Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. Dynamic relu. In *European conference on computer vision*, pages 351–367. Springer, 2020.
- [6] Arnaud De Myttenaere, Boris Golden, Bénédicte Le Grand, and Fabrice Rossi. Mean absolute percentage error for regression models. *Neurocomputing*, 192:38–48, 2016.
- [7] Xinghao Du, Jinhao Meng, Yassine Amirat, Fei Gao, and Mohamed Benbouzid. Exploring impedance spectrum for lithium-ion batteries diagnosis and prognosis: A comprehensive review. *Journal of Energy Chemistry*, 2024.

- [8] Bruce Dunn, Haresh Kamath, and Jean-Marie Tarascon. Electrical energy storage for the grid: a battery of choices. *Science*, 334(6058):928–935, 2011.
- [9] M Elangovan, S Babu Devasenapati, NR Sakthivel, and KI Ramachandran. Evaluation of expert system for condition monitoring of a single point cutting tool using principle component analysis and decision tree algorithm. *Expert Systems with Applications*, 38(4):4450–4459, 2011.
- [10] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35, 2016.
- [11] John B Goodenough and Kyu-Sung Park. The li-ion rechargeable battery: a perspective. *Journal of the American Chemical Society*, 135(4):1167–1176, 2013.
- [12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.
- [13] Alex Graves and Alex Graves. Long short-term memory. *Supervised sequence labelling with recurrent neural networks*, pages 37–45, 2012.
- [14] Ruohan Guo, Feng Wang, M Akbar Rhamdhani, Yiming Xu, and Weixiang Shen. Managing the surge: a comprehensive review of the entire disposal framework for retired lithium-ion batteries from electric vehicles. *Journal of Energy Chemistry*, 2024.
- [15] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.
- [16] Rasheed Ibraheem, Calum Strange, and Goncalo Dos Reis. Capacity and internal resistance of lithium-ion batteries: Full degradation curve prediction from voltage response at constant current at discharge. *Journal of Power Sources*, 556:232477, 2023.

- [17] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.
- [18] Herbert Jaeger. Echo state network. *scholarpedia*, 2(9):2330, 2007.
- [19] Alireza M Javid, Sandipan Das, Mikael Skoglund, and Saikat Chatterjee. A relu dense layer to improve the performance of neural networks. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2810–2814. IEEE, 2021.
- [20] BI Justusson. Median filtering: Statistical properties. *Two-dimensional digital signal prcessing II: transforms and median filters*, pages 161–196, 2006.
- [21] Gholamreza Karimi and Xianguo Li. Thermal management of lithium-ion batteries for electric vehicles. *International Journal of Energy Research*, 37(1):13–24, 2013.
- [22] Serkan Kiranyaz, Onur Avci, Osama Abdeljaber, Turker Ince, Moncef Gabbouj, and Daniel J Inman. 1d convolutional neural networks and applications: A survey. *Mechanical systems and signal processing*, 151:107398, 2021.
- [23] Dominique Larcher and Jean-Marie Tarascon. Towards greener and more sustainable batteries for electrical energy storage. *Nature chemistry*, 7(1):19–29, 2015.
- [24] Xingjun Li, Dan Yu, Vilsen Søren Byg, and Store Daniel Ioan. The development of machine learning-based remaining useful life prediction for lithium-ion batteries. *Journal of Energy Chemistry*, 82:103–121, 2023.
- [25] Junyuan Liang, Hui Liu, and Ning-Cong Xiao. A hybrid approach based on neural network and double exponential model for remaining useful life prediction. *Expert Systems with Applications*, page 123563, 2024.
- [26] Yunpeng Liu, Bo Hou, Moin Ahmed, Zhiyu Mao, Jiangtao Feng, and Zhongwei Chen. A hybrid deep learning approach for remaining useful life prediction of lithium-ion batteries based on discharging fragments. *Applied Energy*, 358:122555, 2024.

- [27] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.
- [28] Kai Luo, Xiang Chen, Huiru Zheng, and Zhicong Shi. A review of deep learning approach to predicting the state of health and state of charge of lithium-ion batteries. *Journal of Energy Chemistry*, 74:159–173, 2022.
- [29] Guijun Ma, Zidong Wang, Weibo Liu, Jingzhong Fang, Yong Zhang, Han Ding, and Ye Yuan. A two-stage integrated method for early prediction of remaining useful life of lithium-ion batteries. *Knowledge-Based Systems*, 259:110012, 2023.
- [30] Guijun Ma, Songpei Xu, Benben Jiang, Cheng Cheng, Xin Yang, Yue Shen, Tao Yang, Yunhui Huang, Han Ding, and Ye Yuan. Real-time personalized health status prediction of lithium-ion batteries using deep transfer learning. *Energy & Environmental Science*, 15(10):4083–4094, 2022.
- [31] Liang Ma, Jinpeng Tian, Tieling Zhang, Qinghua Guo, and Chunsheng Hu. Accurate and efficient remaining useful life prediction of batteries enabled by physics-informed machine learning. *Journal of Energy Chemistry*, 91:512–521, 2024.
- [32] NASA. Li-ion battery aging datasets. https://data.nasa.gov/dataset/Li-ion-Battery-Aging-Datasets/uj5r-zjdb/about_data, Year.
- [33] Christophe Pillot. The rechargeable battery market and main trends 2018–2030. In *36th annual international battery seminar & exhibit. avicenne energy*, page 35, 2019.
- [34] Jiantao Qu, Feng Liu, Yuxiang Ma, and Jiaming Fan. A neural-network-based method for rul prediction and soh monitoring of lithium-ion battery. *IEEE access*, 7:87178–87191, 2019.
- [35] Akhter Mohiuddin Rather, Arun Agarwal, and Vinjamuri Narshima Sastri. Recurrent neural network and a hybrid model for prediction of stock returns. *Expert Systems with Applications*, 42(6):3234–3241, 2015.
- [36] Bhaskar Saha and Kai Goebel. Battery data set. *NASA AMES prognostics data repository*, 2007.

- [37] Ivan Sanz-Gorrachategui, Pablo Pastor-Flores, Milutin Pajovic, Ye Wang, Philip V Orlik, Carlos Bernal-Ruiz, Antonio Bono-Nuez, and Jesús Sergio Artal-Sevil. Remaining useful life estimation for lfp cells in second-life applications. *IEEE Transactions on Instrumentation and Measurement*, 70:1–10, 2021.
- [38] Kristen A Severson, Peter M Attia, Norman Jin, Nicholas Perkins, Benben Jiang, Zi Yang, Michael H Chen, Muratahan Aykol, Patrick K Herring, Dimitrios Fraggedakis, et al. Data-driven prediction of battery cycle life before capacity degradation. *Nature Energy*, 4(5):383–391, 2019.
- [39] V Sugumaran and KI Ramachandran. Effect of number of features on classification of roller bearing faults using svm and psvm. *Expert Systems with Applications*, 38(4):4088–4096, 2011.
- [40] Arvind Sai Sarathi Vasan, Dinesh Michael Mahadeo, Ravi Doraiswami, Yunhan Huang, and Michael Pecht. Center for advanced life cycle engineering (calce), university of maryland, college park, maryland 20742, usa, 2center for prognostics and system health management, city university of hong kong, 83 tat chee avenue, kowloon, hong kong. *Frontiers in bioscience*, 5:39–71, 2013.
- [41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [42] Rui Wang, Mengmeng Zhu, Xiangwu Zhang, and Hoang Pham. Lithium-ion battery remaining useful life prediction using a two-phase degradation model with a dynamic change point. *Journal of Energy Storage*, 59:106457, 2023.
- [43] Shuai Wang, Hongyan Ma, Yingda Zhang, Shengyan Li, and Wei He. Remaining useful life prediction method of lithium-ion batteries is based on variational modal decomposition and deep learning integrated approach. *Energy*, 282:128984, 2023.
- [44] Zhou Wang and Alan C Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE signal processing magazine*, 26(1):98–117, 2009.

- [45] Cort J Willmott and Kenji Matsuura. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate research*, 30(1):79–82, 2005.
- [46] Junyi Xia, Qionglin Shi, Haomiao Li, Min Zhou, Wei Wang, Kangli Wang, and Kai Jiang. Historical data-independent remaining useful life prediction method based on dual-input deep learning neural network. *Journal of Energy Storage*, 72:108427, 2023.
- [47] Zhuang Ye and Jianbo Yu. State-of-health estimation for lithium-ion batteries using domain adversarial transfer learning. *IEEE Transactions on Power Electronics*, 37(3):3528–3543, 2021.
- [48] Xinyou Yin, JAN Goudriaan, Egbert A Lantinga, JAN Vos, and Huub J Spiertz. A flexible sigmoid function of determinate growth. *Annals of botany*, 91(3):361–371, 2003.
- [49] Guangxu Zhang, Xuezhe Wei, Siqi Chen, Gang Wei, Jiangong Zhu, Xueyuan Wang, Guangshuai Han, and Haifeng Dai. Research on the impact of high-temperature aging on the thermal safety of lithium-ion batteries. *Journal of Energy Chemistry*, 87:378–389, 2023.
- [50] Jianqi Zhang, Tao Fan, Shuai Yuan, Chongye Chang, Kuo Wang, Ziwei Song, and Xinming Qian. Patent-based technological developments and surfactants application of lithium-ion batteries fire-extinguishing agent. *Journal of Energy Chemistry*, 2023.
- [51] Yixing Zhang, Fei Feng, Shunli Wang, Jinhao Meng, Jiale Xie, Rui Ling, Hongpeng Yin, Ke Zhang, and Yi Chai. Joint nonlinear-drift-driven wiener process-markov chain degradation switching model for adaptive online predicting lithium-ion battery remaining useful life. *Applied Energy*, 341:121043, 2023.
- [52] Yapeng Zhou and Miaohua Huang. Lithium-ion batteries remaining useful life prediction based on a mixture of empirical mode decomposition and arima model. *Microelectronics Reliability*, 65:265–273, 2016.
- [53] Jiangong Zhu, Yixiu Wang, Yuan Huang, R Bhushan Gopaluni, Yankai Cao, Michael Heere, Martin J Mühlbauer, Liuda Mereacre, Haifeng Dai, Xinhua Liu, et al. Data-driven capacity estimation of commercial

lithium-ion batteries from voltage relaxation. *Nature communications*,
13(1):2261, 2022.