FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 13, Issue 3 | Pages 501-514 | e-ISSN: 2347-470X

Remaining Useful Life Prediction of EV Lithium-Ion Batteries Using TG-AMF Enhanced MTS-BiLSTM Optimized with Walrus Algorithm

Mandeddu Sudhakar Reddy[1*]**, M. Monisha**[2]

[1]Research Scholar, Department of Electronics and Communication Engineering, Vels Institute of Science, Technology & Advanced Studies (VISTAS), Chennai, Tamil Nadu, India*

[2]Assistant Professor, Department of Electronics and Communication Engineering, Vels Institute of Science, Technology & Advanced Studies (VISTAS), Chennai, Tamil Nadu, India

***Correspondence:** *mandeddusudhakarreddy@gmail.com*

ABSTRACT- Predictive maintenance is essential to industrial operations, especially in lithium-ion battery systems. Remaining Useful Lifetime (RUL) prediction is the most accurate means of maintaining optimal performance and avoiding unexpected failure in these systems. Noisy sensor data and complex degradation patterns, however, make the task complex and require sophisticated techniques for proper analysis and forecasting. Hence, this manuscript proposes an optimized Electric Vehicle (EV) lithium-ion battery RUL prediction using a multi-time scale bidirectional Long Short-Term Memory (MTS-BiLSTM) Network. Initially, input data is taken from the Remaining Useful Lifetime Prediction Dataset available on Kaggle, containing sensor readings and operational parameters for systems under degradation. It uses the trimmed global adaptive mean filter approach (TG-AMF), which enhances input data quality by removing general noise and preserving the essential feature points related to degradation patterns. A novel MTS-BiLSTM network approach is proposed for predicting RUL by capturing both Proximal correlations and Delayed dependencies in data to model trends accurately. Hyperparameters of the proposed model are optimized via the Walrus Optimizer (WO), which improves prediction and computation overhead. The robustness of the proposed framework is analyzed through benchmarking with performance metrics coefficient of determination (R2), root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), prediction horizon accuracy (PHA), and computation time (CT). The overall PHA of 98%, MAE of 17.04, MAPE of 8.63, CT of 7.57s, and RMSE of 5.83 are obtained by the proposed method of forecasting the battery RUL for EVs.

Keywords: Remaining Useful Lifetime (RUL), Electric Vehicles (EVs), Predictive Maintenance, Lithium-Ion Battery, Deep Learning (DL), Parameter Tuning, Data Preprocessing.

ARTICLE INFORMATION

Author(s): Mandeddu Sudhakar Reddy, M. Monisha;

Received: 24/04/2025; **Accepted**: 07/08/2025; **Published**: 30/09/2025;
E- ISSN: 2347-470X;
Paper Id: IJEER 2404-08;
Citation: 10.37391/ijeer.130315
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-13/ijeer-130315.html

Crossref member
CROSSREF.ORG
THE CITATION LINKING BACKBONE

Publisher's Note: FOREX Publication stays neutral with regard to jurisdictional claims in Published maps and institutional affiliations.

1. INTRODUCTION

Predictive maintenance has emerged as a critical strategy in industrial operations, especially for systems that rely on complex machinery and components such as lithium-ion batteries [1]. As industries continue to integrate advanced technologies into their infrastructure, ensuring the reliability and longevity of these systems has become highly important [2]. Predictions on the Remaining Useful Lifetime (RUL) of components enable timely proactive maintenance, reducing unnecessary downtime and optimizing operational expenses [3]. This is particularly applicable to applications like electric vehicles (EVs), for which battery performance is important in enabling the effective functioning of the system and user satisfaction [4]. This task is not simple: it predicts the RUL of the lithium-ion battery [5]. Different dilapidation mechanisms are inherent in the Li battery and may significantly vary based on usage patterns, environmental conditions, and other factors [6]. Data obtained from the battery system is usually noisy and complex, and may include informative signals and unwanted disturbances [7]. This complexity makes it challenging to model the degradation process accurately and make reliable predictions about when the system will fail or require maintenance [8].

For effective RUL prediction, capturing the Proximal correlations and Delayed dependencies trends in the data that reflect the degradation process [9] [10]. Battery data typically includes time-series measurements such as electrical energy, current, temperature, and state of charge (SoC), which change over time as the battery degrades [11]. In particular, traditional models have difficulty with such data complexities and their inherent noise, nonlinearity, and long-term dependencies in battery behavior [12] [13]. In addition, for many real-time applications, processing a large amount of data must be performed efficiently, which requires computational overhead to be paramount in developing predictive models [14]. With the cumulative demand for reliable and precise RUL forecast systems, there is an increasing trend toward new methodologies that enhance prediction accuracy without increasing computational costs [15] [16]. All these are developed to be able to overcome noisy and complex data challenges with scalability to real-world applications [17]. This balance is critical for industries such as EVs, renewable energy storage, and consumer microchip technology, where battery failure can cause significant operational disruptions and safety risks [18]. Therefore, the development of robust and efficient prediction frameworks is essential for the future of predictive maintenance in these critical sectors [19] [20].

Motivation: Existing works on RUL prediction for EV lithium-ion batteries have made significant strides, but several challenges persist. One key issue is the noisy and incomplete sensor data, which can negatively impact the accuracy of predictions. This problem arises due to sensor malfunctions, communication errors, or environmental factors leading to data corruption or missing values. Additionally, many existing methods rely on simplistic models that fail to capture the complex, nonlinear relationships inherent in the dilapidation process of Li batteries. These models often do not consider the influence of various operational factors such as temperature fluctuations, charging/discharging cycles, and driving conditions, leading to inaccurate or overly generalized predictions. Another common limitation is the lack of accurate prediction capabilities. Furthermore, most of the existing approaches struggle to generalize across different battery types or operational environments, limiting their applicability in real-world scenarios. Hyperparameter optimization in many methods is either not performed or is done inadequately, resulting in suboptimal performance. Moreover, the lack of effective feature engineering and the reliance on raw data without significant preprocessing often leads to models that cannot effectively capture the underlying degradation patterns. Despite these efforts, few approaches focus on integrating deep learning models with optimization algorithms to enhance prediction accuracy and computational efficiency. Motivated by these challenges, the proposed work proposes a more accurate, efficient, and adaptable RUL prediction model that leverages advanced preprocessing, deep learning techniques, and accurate prediction capabilities for improved battery management in EVs.

The foremost contributions of the proposed framework are as follows:

- To introduce a novel multi-time scale bidirectional Long Short-Term Memory (MTS-BiLSTM) Network for accurate RUL prediction. This network effectively captures both Proximal correlations and Delayed dependencies in sensor data, enabling precise modeling of degradation patterns.
- To enhance input data quality using the Trimmed Global Adaptive Mean Filter (TG-AMF) approach, which effectively removes noise while preserving critical feature points related to degradation trends, improving the reliability of the predictions.
- To optimize the hyperparameters of the MTS-BiLSTM network using the Walrus Optimization Algorithm (WOA), achieving improved prediction accuracy and reduced computational overhead compared to conventional optimization methods.
- To validate the robustness of the developed framework through benchmarking against traditional methods using multiple performance metrics, including R^2, RMSE, MAE, MAPE, PHA, and computation time, demonstrating superior performance in all aspects.
- To utilize the RUL Prediction Dataset from Kaggle as a freely accessible dataset for evaluating the proposed framework, ensuring its practical applicability for forecasting the RUL of EV Li batteries.

The forthcoming sections are prearranged as follows: *section 2* emphasizes the related work, *section 3* deliberates over the suggested approaches, *section 4* presents the results and discussion, and *section 5* represents the conclusion of the developed framework.

░ 2. RECENT RESEARCH WORK: A BRIEF REVIEW

This section highlights recent efforts in predicting the RUL of Li batteries using DL techniques for accurate performance forecasting.

In 2024, Swain et.al., [21] have presented an ML-based forecaster for EV batteries' remaining usable life. The purpose of the utilized approach was the prognosis of Lithium-ion Batteries' Remaining Useful Life through advanced ML techniques, more specifically RF and SVM, while applied to the most recent dataset from the NASA Ames Prognostics Center of Excellence. Features are optimized by applying One-way ANOVA, followed by a rich hyperparameter tuning to enhance the precision of a model. Real-time factors like temperature fluctuations and usage cycles were also integrated into the research to study their effects on the performance of the battery. The R2 score and MSE were used to evaluate the introduced models. However, the approach faced limitations in generalizing transversely varied battery types and operative situations.

In 2023, Gao et.al., [22] have suggested a hybrid CNN-BiLSTM method for predicting the EV Li battery's remaining usable life. This uses a 1D CNN-BiLSTM for predicting the RUL of Li-ion batteries in EVs. Using a 1D CNN, it was combined with a BiLSTM and also employs ELU activation functions in the convolutional layer. Thus, the hybrid model would have a better ability to achieve more precise and stable RUL forecasts. The 1D CNN picks deep features from the State of Health (SOH), and the BiLSTM captures patterns in both ways, with the final estimation of RUL through dense layers. For verification and validation of the approach, several comparisons were made using NASA battery data and estimation of performance by the introduced architecture against traditional RNN models, LSTM models, hybrid models, and Bi-LSTM models.

In 2024, Mishra et.al., [23] have developed an automated ML for premature RUL using novel holistic health indicators to improve EV lithium-ion battery management. It introduces new indicators for battery aging characterization using correlation analysis, such as CC incriminating location values, CV incriminating peaks, and

CC discharging. In addition to these new indicators, this study combines them with the traditional health metrics, namely capacity, cycle, cluster, and checkpoint data, for better RUL prediction accuracy. K-means clustering was used for grouping similar data, while the ruptures library was used to detect the change points in the data of the battery. In terms of experimental results on NASA's AMES LIB cycle life dataset, improvements over existing methods were exhibited. However, the proposed approach still had issues handling variability in data across different types of batteries and operational conditions.

In 2023, Zhao et.al., [24] have suggested an ML approach to forecast electric vehicle battery capacity. Here presented feature-based ML methods were presented for the estimation of capacity on Macro-scale LiFePO4 batteries for EV deployment and predicting capacity fade trajectories. There was a massive dataset formed from 420 cells and 9 battery packs, yielding over 10,000 validation data points through a cloud platform. There was then a Bi-level noise reduction technique applied to filter these data, while 39 domain-specific features were engineered using battery charging data. The utilized stacking ensemble learning approach was a combination of four base models and a meta-learner to improve prediction accuracy and generalization. Still, the approach was very limited in scalability when dealing with larger, more diverse datasets.

In 2022, Cai et.al., [25] have developed an EV power battery's remaining practical life forecast technique for DC rapid charging conditions. This paper presented an adaptive approach to predicting the RUL of EV power batteries under direct contemporary dissolute charging circumstances. The method was considered a way to diminish the impact of multifaceted health indicators (HI) on the exactness of predictions. It applied a WSA-LSTM approach that integrates real-time condition monitoring and simulation data through the DDDAS. Pearson and Spearman correlation coefficients combined with the entropy weight process were applied for key feature selection that interferes least with secondary HIs. The effectiveness of both the techniques, feature selection and RUL prediction methods, was proven in the simulation experiments done based on real-time data. However, the approach experienced some difficulty in dealing with large datasets and varying operational conditions.

In 2022, Jafari et.al., [26] have developed a forecast for the health of Li-ion batteries in hybrid vehicles with machine learning. A hybrid framework for the forecast of the SOH of Li-ion batteries was presented in EVs. With increasing concerns about environmental and sustainability issues, the optimal performance and safety of Li-ion batteries require special attention. The research used an XGBoost algorithm to estimate the SOH with error breakdown to enhance the performance parameters. The model was further improved by accuracy correction for SOH estimation by considering key features, such as voltage, current, and temperature differences. However, the approach posed challenges to real-time adaptability in dynamic charging conditions and larger datasets.

In 2025, Ma et.al., [27] have introduced a semi-supervised representation learning method to enhance prediction accuracy by leveraging data without RUL labels. The framework was built upon a sophisticated deep neural network architecture consisting of an encoder and three decoder heads, designed to extract time-dependent representation features from short-term battery operating data, irrespective of the presence of RUL labels. The effectiveness of the approach was validated using three datasets obtained from 34 batteries operating under diverse conditions, covering more than 19,900 charge and discharge cycles. Despite its promising performance, the method exhibited certain limitations. Specifically, its effectiveness was influenced by the quality and diversity of the unlabeled data, and the generalization capability across unseen battery chemistries and extreme operating scenarios remained constrained. Additionally, the model's interpretability was limited, making it challenging to gain clear insights into the learned representations and decision processes.

3. PROPOSED METHODOLOGY

This manuscript proposes an optimized EV's lithium-ion battery RUL prediction using an MTS-BiLSTM Network. *Figure 1* indicates the workflow of the proposed framework.

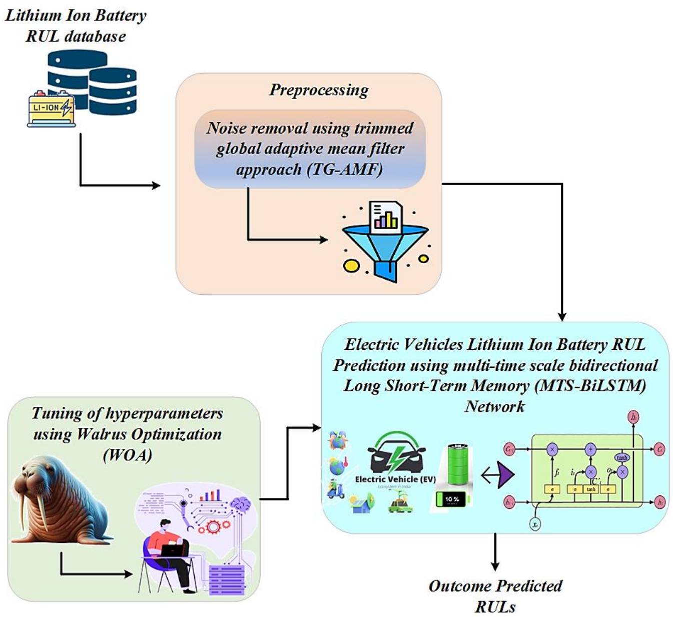
Lithium Ion Battery RUL database
LI-ION
Preprocessing
Noise removal using trimmed global adaptive mean filter approach (TG-AMF)
Electric Vehicles Lithium Ion Battery RUL Prediction using multi-time scale bidirectional Long Short-Term Memory (MTS-BiLSTM) Network
Electric Vehicle (EV)
10 %
Tuning of hyperparameters using Walrus Optimization (WOA)
Outcome Predicted RULs

Figure 1. Workflow of the Proposed Framework

Initially, input data is taken from the Remaining Useful Lifetime Prediction Dataset available on Kaggle, containing sensor readings and operational parameters for systems under degradation. It uses the TG-AMF, which enhances the quality of input data by removing noise in general and preserving the essential feature points related to degradation patterns. A novel MTS-BiLSTM network approach is proposed for predicting RUL by capturing both Proximal correlations and Delayed dependencies in data to accurately model trends. Hyperparameters of the proposed model are optimized via the WOA, which brings improvement in prediction and computation overhead.

3.1. Data Acquisition

The RUL dataset [28] consists of 15,064 samples with 9 features, primarily focusing on the performance characteristics of batteries and their Remaining Useful Life (RUL). The features include Cycle-Index, which represents the battery's cycle count, and Discharge Time (DT) (s), indicating the time taken during the discharge phase in seconds. Another feature, Decrement 3.6-3.4V (s), measures the duration spent within the voltage range of 3.6V to 3.4V during discharge. Voltage-related features include Max. Voltage Discharge. (V), capturing the maximum voltage during discharge and the minimum. Voltage Charge (V), which notes the minimum voltage during charging. Additional charging-related parameters include Time at 4.15V (s), denoting the duration at 4.15V during charging, Time constant current (s), representing the time at constant current, and Charging time (s), the total charging time in seconds. Finally, the RUL feature provides the battery with an RUL in cycles. This dataset is structured to enable comprehensive analysis and prediction of battery health and performance over its lifecycle.

3.2. Preprocessing Stage

The raw samples collected from the freely accessible source contain a high level of irrelevant noises that minimize the model's prediction performance. Nowadays, the mean filter (MF) [29] approach effectively eliminates unwanted feature points from the data instances. However, the traditional MF technique requires sorting the values in a sliding window, which can be processor-intensive for huge datasets or real-time applications. To overcome this issue, a trimmed global adaptive mean filter approach (TG-AMF) is introduced to avoid over-smoothing and helps in retaining critical features such as edges in images or sharp transitions in time-series data.

The proposed filtering technique is specifically applied to noisy samples. This filtering method identifies the irrelevant data using the rank-ordered absolute difference (ROAD), and a standardized threshold value is set in the initial stage. ROAD determines the noisy instances by contemplating the variance between the data considered and the adjacent data subsequently. The data is considered to be noise-free only when the ROAD value is less than the defined threshold value, or otherwise considered to be irrelevant data. The binary data is determined after the identification of noisy samples; 1 represents the cleaned data, and 0 represents the noisy data. Element-wise product of noisy and binary data determines the outcome samples. The pixel distance was analyzed using the ROAD statistics using the adjacent feature values. Assume the data location as $i(i_1, i_2)$ and $(2M + 1) \times (2M + 1)$ indicates the window size that is generated and centered at i, whereas Mindicates a positive integer. Detection window size $w \times w$is indicated as, W. The set $\beta_i(M)$ can be mathematically formulated as,

$$\beta_z(M) = \{z + (x, y); -M \leq x, y \leq M\} \tag{1}$$

Assume $M = 1$, and $\beta_i^0(M)$ indicates the set of points in a 3×3 identified the adjacent of $i$can be mathematically formulated as,

$$\beta_z(M) = \beta_z(1)/\{z\} \tag{2}$$

Here, n represents the predefined threshold value, and $c_x(z)$indicates the minimal distance $d_{i,j}$for $j \in \beta_i^0, M = 1$ and $n = 4$. The data intensity with the absolute difference can be mathematically formulated as,

$$d_{i,j} = |v_i - v_j| \tag{3}$$

Arrange $d_{i,j}$values in cumulative order and determine the ROAD using the equation below:

$$ROAD_n(i) = \sum_{x=1}^{n} c_x(i) \tag{4}$$

After arranging, the value of $n$is initialized. If the ROAD value surpasses the threshold value, it is determined as noisy data; otherwise, it is considered noiseless. The noisy and noise-free data are separated at this stage using binary samples. Generated binary samples b can be formulated as,

$$b(x, y) = \begin{cases} 0 for ROAD(x, y) < R \\ 1 for ROAD(x, y) > R \end{cases} \tag{5}$$

The outcome data obtained for upcoming processing is determined using element-wise multiplication if input samples and actual data, which can be formulated as,

$$P(x, y) = I(x, y) \times b(x, y) \tag{6}$$

Here, $P(x, y)$ indicates the outcome samples, $I(x, y)$ represents the actual data, and $b(x, y)$symbolizes the binary samples. The binary samples are generated by evaluating the distance between data features and distinguishing them with the threshold value. The detected irrelevant information identified in the earlier phase is then considered and replaced using the proposed TG-AMF technique. Here, the filtered window size is maximized using the irrelevancy in the adjacent samples. The TG-AMF Mcan be formulated using *equation (7)*,

$$N = \frac{1}{M} \sum_{x \in M} \hat{f}(x) \tag{7}$$

Here, Mindicates the noise-free data, and $\hat{f}$signifies the noise-free features in the samples. After the filtering process, data normalization is performed using the min-max normalization technique. This process can improve the efficacy and robustness of the developed scheme. Here, the min-max normalization (MMN) technique is introduced and can be mathematically formulated in *equation (6)*,

$$Z_{normalized}(a, b) = \frac{Z_{a,b} - Z_{min}}{Zmin_{max}}$$

Here, $Z_{a,b}$indicates the Z_{min}indicates the minimum value, and Z_{max}represents the minimum value. Algorithm 1 depicts the pseudocode for the TG-AMF technique

#Algorithm 1: Pseudocode for TG-AMF technique

```
Input:
  D ← Raw input dataset with noisy samples

  W ← Window size (odd integer)

  T ← Predefined threshold for ROAD (e.g., 10% of max
variation)
```

```
Output:
  $D_{filtered}$ ← Preprocessed data with reduced noise

For each data point $x_i$ in $D$ do
    Extract a local window $W_i$ centered around xi
    Compute $ROAD(i)$ ← mean of absolute differences between xi and its neighbors in Wi
    If $ROAD(i) > T$ then
        Mark $x_i$ as noisy $(n_i = 0)$
    Else
        Mark $x_i$ as clean $(n_i = 1)$
    End If
End For
Create binary mask $N = \{n_1, n_2, \ldots, nN\}$
Compute filtered data:
    $D_{filtered} = D \times N + TG - AMF(D, W)$
    // Replace noisy values with adaptive mean from trimmed neighbors
Normalize $D_{filtered}$using Min-Max normalization

Return $D_{filtered}$
```

3.3. RUL Prediction using MTS-BiLSTM Network

The preprocessed data is then fed into the proposed multi-time scale bidirectional Long Short-Term Memory (MTS-BiLSTM) model to extract the spatiotemporal patterns and accurately predict the RUL of the Li-ion battery. Since the traditional LSTM model [30] suffers due to the gradient exploitation problem, a novel multi-time scale function is proposed that prevents over-fitting issues during the training process. Various temporal features like Cycle-Index, (DT), Decrement voltage, maximum and minimum VD, time constant current, charging time, etc. The detailed analysis of the proposed framework is discussed below: The LSTM is a diverse subtype of RNNs utilized for progressive data processing. The LSTM consists of two essential components: the gating unit (GU) and memory unit (MU). The MU helps to store and transfer data, whereas the GU maintains the data flow regulation. The input gates (IG) deliberate on the concern over transmitting new input into the MU. Simultaneously, the *tanh* function is computed to obtain vectors, thereby upgrading the parameters within the MU. By utilizing the sigmoid function and previous memory states, this gate produces a value between the range 0 and 1. The outcome gate is assigned to discriminate the memory cell segments to send the corresponding hidden state. The mathematical expressions for the LSTM training process are deliberated in *equations (8-12)*,

$$x_k = \sigma(W_{ix} i_k + W_{ih} h_{k-1} + W_{id} d_{k-1} + Bias_x), \tag{8}$$

$$f_k = \sigma(W_{if} i_k + W_{hf} h_{k-1} + W_{fd} d_{k-1} + Bias_f), \tag{9}$$

$$d_k = f_k d_{k-1} + x_k \tanh(W_{id} i_k + W_{hd} h_{k-1} + Bias_d), \tag{10}$$

$$y_k = \sigma(W_{yx} i_k + W_{yh} h_{k-1} + W_{yd} d_k + Bias_y), \tag{11}$$

$$h_k = y_k \tanh(d_k) \tag{12}$$

Here, σ indicates the sigmoid function, x, f, d, and $y$represents the IG, forget gate, activated cell vector, and output gate, respectively. Moreover, h and Wrepresents the hidden trajectory, and the weights between two units. However, there arises some complexity in accessing previous state information using the LSTM model. To tackle this issue, the MST-LSTM model utilizes both forward and backward LSTMs to process the incoming features in opposite directions. This duo-directed technique optimally controls the signals based on the current time step. This wide-ranging perception assists in determining active patterns and bears the long-term dependencies over fault samples. Initializing the training process, the sequential feature subsets are encompassed in the MTS-BiLSTM model. The sequence lengths are coordinated at the input layer to ensure uniformity. This process aids the sequential inputs with similar time steps, resulting in $y_{1,} y_2, \ldots . . y_n$ have similar time steps after being utilized as the input. The flow of samples regulates the memory updates via the LSTM layer, deliberating MTS-BiLSTM to integrate both backward and forward directions. *Figure 2* indicates the architecture of the MTS-BiLSTM Technique.

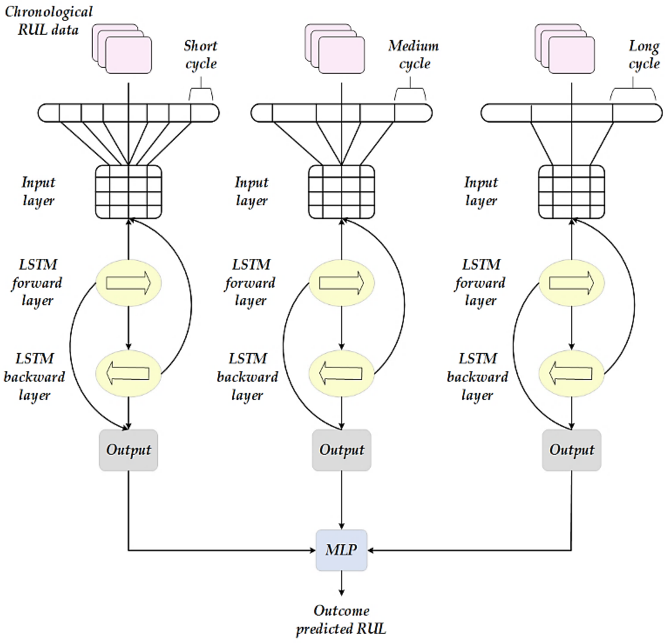

Figure 2. Architecture of MTS-BiLSTM Technique

To account for the varying characteristics of battery degradation data across different time scales, tailored BiLSTM models are applied to the battery RUL dataset. To capture the varying characteristics of battery degradation trends across multiple temporal depths, the proposed MTS-BiLSTM model utilizes a multi-branch architecture, where each branch processes data over different *cycle-based lookback windows*. Specifically, three BiLSTM models are applied independently using short-term, medium-term, and long-term historical sequences—for instance, over the last 10, 25, and 50 charge-discharge cycles. Each model takes input features such as discharge time $(t_{\text{k-1}}, t_{k-2}, \ldots, t_{k-m})$, voltage decrements $(v_{\text{k-1}}, v_{k-2}, \ldots, v_{k-m})$, time spent at 4.15V $\left(t_{\text{4.15V,k-1}}, t_{4.15V,k-2}, \ldots, t_{4.15V,k-m}\right)$, maximum voltage during discharge $v_{max,k}$, and minimum charge voltage $\left(v_{min,k-1}, \ldots., v_{min,k-m}\right)$. Here, m denotes the lookback window size corresponding to each BiLSTM branch. All sequences are derived solely from cycle-level observations. Each branch independently learns degradation patterns relevant to its respective window size, and the outputs are later combined to form a more robust and accurate RUL prediction. A stacking approach enhances prediction stability and practical application during the ensemble stage. Combining outputs from the three LSTM sub-models achieves this by using a multilayer perceptron (MLP). The MLP synthesizes predictions, with its hidden layer output $h_k$reflecting the estimated RUL $r_t$for each sub-model. Hyperparameters like learning rate, activation function, number of layers, and neurons per layer are optimized based on MLP validation performance. The MLP's input dimension is set to 3, corresponding to regression outputs from the three BiLSTM sub-models. Its output dimension is 1, which represents the final RUL prediction. This combined output forms the MTS-BiLSTM model's prediction result, integrating the three sub-models for accurate RUL forecasting in EV battery systems.

3.4. Parameter Tuning using WOA Technique

The proposed MTS-BiLSTM technique causes high complexity while training with larger data. This may lead to the loss of essential features and subject to increased error. To overcome this issue, parameters like batch size, learning rate, epochs, and dropout rates are tuned before providing the data to the proposed network model. Metaheuristic optimizers update the model parameters with a globally optimal solution to perform this. The proposed framework introduced a Walrus Optimization Algorithm (WOA) [31] to tune the weight parameters of the network model. The WOA takes a holistic approach to solving optimization problems, drawing inspiration from walrus behaviors during feeding, migration, and predator avoidance. WOA improves global search capabilities by exhaustively searching the solution space and mimicking walrus migration behaviors toward more favorable settings. WOA simulates the intelligent foraging, migration, and predator avoidance behaviors of walruses. Each walrus represents a candidate solution in the search space, with its position encoding a possible set of hyperparameters. In order to ensure convergence to the best results, it fine-tunes solutions locally during the exploitation phase, mimicking how walruses modify their movements to obtain food or avoid danger.

Large marine mammals that are mostly found in the cold waters of the Northern Hemisphere, walruses are easily recognized by their characteristic whiskers and tusks. Both males and females have these extended canine teeth, or tusks, which can reach a length of one meter and weigh about 5.4 kg. Typically, walruses graze on benthic bivalve mollusks found on ocean ice. Walruses frequently move toward rocky beaches or remote locations when temperatures rise and sea ice starts to melt. Large crowds and a lot of activity are characteristics of these movements. Because of their size and power, killer whales and polar bears are walruses' main natural predators. Nevertheless, walruses demonstrate remarkable intellect in their daily routines and social relationships. Using their tusks to guide others to food sites, moving onto rocky coasts during warmer seasons, and protecting themselves against predators through flight or combat are the three main tasks that best demonstrate their cunning behavior.

Step 1: Initialization Phase

One of the most important steps in this phase is initializing the walrus population within the search space. In order to avoid convergence to local optima, this procedure guarantees a varied distribution throughout the search space. The search area is made more varied by this stage, where each walrus represents a possible solution. The walruses serve as the search agents in the WOA, and each one represents a potential fix for the optimization issue. Each walrus's location in the search space correlates to certain values of the variables related to the answer. When the WOA procedure begins, the walrus population is dispersed at random. The population matrix for WOA is obtained using *equation (13)*.

$$Y = \begin{bmatrix} Y_1 \\ .. \\ Y_u \\ .. \\ Y_M \end{bmatrix} = \begin{bmatrix} y_{1,1} \cdots y_{1,v} \cdots y_{1,n} \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ y_{u,1} \cdots y_{u,v} \cdots y_{u,n} \\ \cdots\cdots\cdots\cdots\cdots \\ y_{M,1} \cdots y_{M,v} \cdots y_{M,n} \end{bmatrix} \tag{13}$$

Here, Y indicates the walrus population, $n$represents the number of chosen variables, Mindicates the total walruses, $y_{u,v}$indicates the v^{th}dimension presented by u^{th}walrus, and $Y_u$represents the u^{th}walrus.

Step 2: Random Generation

After the initialization, randomly choose the most appropriate solution from the set of input parameters.

Step 3: Fitness Function

The WOA utilized fitness function (FF) for analyzing the optimality of the proposed classifier model, and it is mathematically formulated in *equation (14)*,

$$f = min(errorrate) \tag{14}$$

Step 4: Exploration Phase

To concentrate their search inside a small area, walruses naturally shift positions somewhat during this phase, simulating their method of avoiding predators. Similarly, this stage of the algorithm refines the solution to get closer and closer to the global optimal solution. Walruses make exploratory moves to diversify the population, avoiding local optima. Position updates simulate foraging and group-guided movement. Numerous kinds of marine life are among the many marine species that walruses consume. They forage along the ocean floor to find the benthic bivalve mollusks, mostly shellfish, that make up their preferred diet. They find food by using their sensitive touch senses and nimble flipper motions. The group's leading walrus, identified by its massive tusks, guides the others in their hunt for food. The placements of the walruses are changed according to their grazing behavior *using equations (15) and (16)*.

$$y^{L1}_{u,v} = y_{u,v} + rand_{u,v}\left(sw_v - Z_{u,v}.y_{u,v}\right) \tag{15}$$

$$Y_u = \begin{cases} Y^{L_1}_u, f^{L_1}_u < f_u \\ Y_u, else \end{cases} \tag{16}$$

Here, $y^{L1}_{u,v}$indicates the upgraded position for u^{th}walrus based on feeding, $y^{L1}_{u,v}$indicates the v^{th}dimension, $f^{L_1}_u$represents the fitness value, $rand_{u,v}$indicates the arbitrarily generated number, which ranges from 0 to 1. sw indicates the likelihood solution is connected to the magnificent walrus, and $Z_{u,v}$indicates the number designated randomly between 1 and 2. $Z_{u,v}$is implemented to enhance the method's search capability, hence it is represented as 2. It creates more enhancements and complex dissimilarities in the individual's location. The value is chosen as 1, which comprises a regular phase of this movement.

Step 5: Migration Phase

The walrus' behavior contemplates migrating based on the temperature variations determines the inspiration for this stage, which excites the different exploration points within the search space. Inspired by seasonal relocation, walruses shift toward optimal areas in the search space to refine candidate solutions. For this case, the objective contemplates the search for secluded positions to prevent the optimizer from facing the local optima solution. At the summer search in increasing temperature, the likelihood behavior of walruses defines the process of propagating adjacent to the outcrops. The WOA performs the migration behavior to linear the walrus to identify a suitable location within the search space. Based on the below *equations (17) and (18)*, the most present position will thrive the existing walrus if enhances the fitness value.

$$y^{L2}_{u,v} = \begin{cases} y_{u,v} + rand_{u,v}.\left(y_{p,v} - Z_{u,v}.y_{u,v}\right), f_p < f_v \\ y_{u,v} + rand_{u,v}.\left(y_{u,v} - y_{p,v}\right), else \end{cases} \tag{17}$$

$$Y_u = \begin{cases} Y^{L_2}_u, f^{L_2}_u < f_u \\ Y_u, else \end{cases} \tag{18}$$

Here, $Y^{L_2}_u$indicates the upgraded position for u^{th} walrus based on migration, $y^{L2}_{u,v}$ indicates the v^{th} dimension, $f_p$indicates fitness values, $f^{L_2}_u$ indicates fitness values, $Y_p, p \in \{1,2,\ldots.,M\}$ and $p \neq u$symbolizes the pointed walrus adjacent u^{th}walrus and $y_{p,v}$indicates v^{th} dimension.

Step 6: Exploitation Phase

Predators like polar bears and killer whales pose serious hazards to walruses, which frequently forces them to travel constantly across their environment to avoid and deal with these threats. An algorithm's capacity to investigate condensed search areas surrounding possible solutions can be enhanced by drawing inspiration from this natural behavior. It fine-tunes the solutions by narrowing the local search region, simulating predator avoidance and focused foraging. The dynamic range is reduced over iterations to converge to the global optimum. Each walrus's adjustment process takes place within a specific neighborhood, the range of which is first set to its greatest value and then shrinks while the algorithm runs. Finding the best areas in the search space requires a great deal of investigation in the early iterations. This is made possible by the introduction of a dynamic range, which modifies the boundaries for localized searches as part of the WOA method. This dynamic modification mimics the walrus's spatial distribution, in which every individual is encircled by a region that symbolizes its immediate surroundings. *Equations (19) and (20)* customized to the algorithm's specifications are then used to generate a new random point inside this localized region.

$$y^{L_3}_{u,v} = y_{u,v} + \left(lb^t_{local,v} + \left(ub^t_{local,v} - rand.lb^t_{local,v}\right)\right) \tag{19}$$

$$LB\!:\begin{Bmatrix} lb^t_{local,v} = \frac{lb_v}{t} \\ ub^t_{local,v} = \frac{ub_v}{t} \end{Bmatrix} \tag{20}$$

$$Y_u = \begin{cases} Y^{L3}_u, f^{L_3}_u < f_u \\ Y_u, else \end{cases} \tag{21}$$

Here, $ub^t_{local,v}$and $lb^t_{local,v}$deliberates the local lower bounds for v^{th}variable. $f^{L_3}_u$indicates the fitness value, Y^{L3}_uindicates the v^{th}dimension, $ub_v$and lb_vsignifies upper and lower bounds.

Step 7: Return the Best Optimal Solution

Step 8: Termination

Finally, the parameters of the MTS-BiLSTM technique are tuned using the WOA, repeating step 3 until the $k = k + 1$ stopping criteria are met. *Figure 3* depicts the flowchart of the developed WOA technique. *Algorithm 2* represents the pseudocode of the proposed WOA technique.

#Algorithm 2: Pseudocode of the Proposed WOA Technique

```
Begin WOA
    Define the optimization problem and the relevant objective function.
    Initialize parameters: number of walruses M, and maximum iterations T
    Randomly assign initial positions to all walruses in the search space.
    For each iteration $t = 1$ to $T$:
        Identify the walrus with the best fitness value (strongest candidate).
        For each walrus $u = 1$ to$M$:
```

```
        Phase 1: Feeding Behavior (Exploration)
            Compute a new exploratory position for the walrus v using equation (15).
            Update the walrus ulocation using equation (16)
        Phase 2: Migration
            Select a migration target for the walrus u.
        Determine the new location using equation (17).
        Adjust walrus uposition based on equation (18).
        Phase 3: Escaping/Fighting Predators (Exploitation)
Generate a nearby candidate solution using equations (19) and (20).
    Update the location of the walrus uusing equation (21).
  End For
      Retain the best solution found in this iteration if it outperforms previous candidates
End For
    Return the best solution (quasi-optimal) identified by WOA for the defined problem
End WOA
```

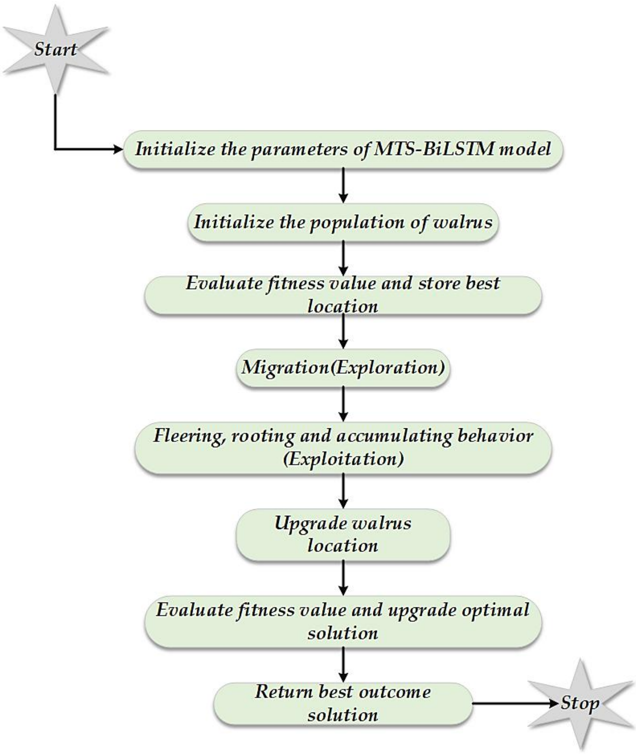
Start
Initialize the parameters of MTS-BiLSTM model
Initialize the population of walrus
Evaluate fitness value and store best location
Migration(Exploration)
Fleering, rooting and accumulating behavior (Exploitation)
Upgrade walrus location
Evaluate fitness value and upgrade optimal solution
Return best outcome solution
Stop

Figure 3. Flowchart of the developed WOA Technique

4. RESULTS AND DISCUSSION

The proposed scheme is analyzed and processed *via* the Python simulation platform. For the experimentation process, hyperparameters like learning rate, min-batch size, dropout, and several epochs are considered. The proposed method is processed under Intel(R) Core (TM) i5-4300M CPU with 4GB installed RAM using a 64-bit operating system. For the training process, 80% of the training data, 10% of the testing data, and 10% of the validation data are considered, which are in the ratio 8:1:1. *Table 1* indicates the Hyperparameters of the Developed Method.

Table 1. Hyperparameters of the Developed Method

Parameters Used	Values
Learning rate	0.001
Batch size	64
Dropout	0.3
Total number of epochs	100
Optimizer	WOA
Search Agents	30
Maximum Iteration	100
Fixed-length time window	30
Input Shape for BiLSTM	30×9 matrix
Data normalization	Min-max; range [0,1]

4.1. Assessment Measures

To better understand the proposed approach, performance indicators R2, RMSE, MAE, MAPE, prediction PHA, and CT are computed.

4.1.1. R-squared Analysis

R^2 measures how well the predicted values explain the variability of the actual RUL data. It spans from 0 to 1, with values nearer to 1 reflecting a better fit. It can be evaluated using *equation (22)*,

$$R^2 = 1 - \frac{\sum_{x=1}^{M}(u_x-\hat{u}_x)^2}{\sum_{x=1}^{M}(u_x-\tilde{u}_x)^2} \qquad (22)$$

Here, $u_x$signifies the actual RUL, $\hat{u}_x$defines the predicted RUL, $\tilde{u}_x$contemplates the mean of actual RUL values, and Msignifies the number of samples.

4.1.2. RMSE Analysis

MSE quantifies the average magnitude of errors between definite and forecast RUL values. A lower RMSE indicates better predictive performance. It can be evaluated using *equation (23)*,

$$RMSE = \sqrt{\frac{1}{M}\sum_{x=1}^{M}(u_x-\hat{u}_x)^2} \qquad (23)$$

4.1.3. MAE Analysis

MAE calculates the average of absolute differences between definite and forecast RUL values, providing an intuitive measure of prediction error. It can be evaluated using *equation (24)*,

$$MAE = \frac{1}{M}\sum_{x=1}^{M}|u_x-\hat{u}_x|^2 \qquad (24)$$

4.1.4. MAPE Analysis

MAPE represents prediction error as a percentage, making it useful for comparing errors across different scales. It can be evaluated using *equation (25)*,

$$MAPE = \frac{1}{M}\sum_{x=1}^{M}\left|\frac{u_x-\hat{u}_x}{u_x}\right| \times 100 \qquad (25)$$

4.1.5. PHA Analysis

PHA evaluates how accurately the model predicts the RUL within a specified tolerance window near the end of life (EOL). It can be evaluated using *equation (26)*,

$$PHA = \frac{Number\ of\ predictions\ within\ tolerance\ window}{Total\ Pr\ ediction} \times 100 \quad (26)$$

Tolerance Window: Define acceptable deviation from the true RUL, *e.g.*, ±10%.

4.2. Correlation Matrix Analysis

In this section, the Correlation matrix analysis for the input data is analyzed under different feature subsets. This correlation matrix plot visually represents the relationships between various features related to battery parameters and Remaining Useful Life (RUL). The values in the matrix range from -1 to 1, where 1 indicates a perfect positive correlation, -1 indicates a perfect negative correlation, and 0 indicates no correlation. The plot uses both colors and circular segment sizes to indicate the strength and direction of the correlations.

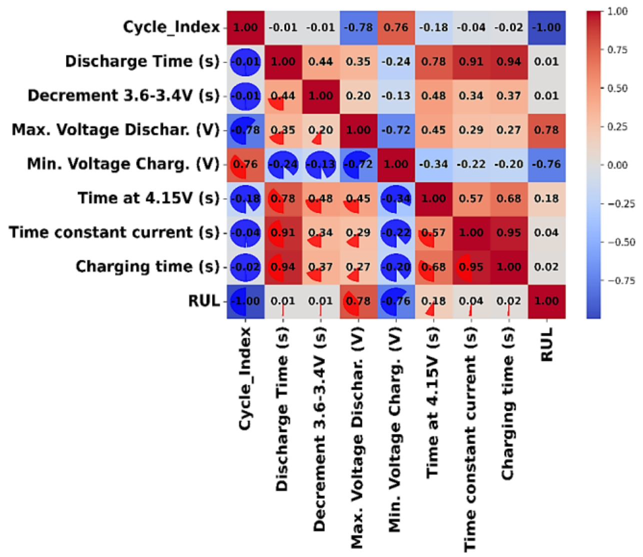

Figure 4. Correlation Plot Analysis of the Input Data

Figure 4 signifies the correlation plot analysis of the input data. From the matrix, we observe that the "Cycle_Index" and "RUL" exhibit a perfect negative correlation (-1.00), meaning as the cycle index increases, the RUL decreases consistently. Similarly, "Min. Voltage Charg. (V)" is negatively correlated with both "Cycle_Index" (-0.76) and "RUL" (-0.76), suggesting its significant role in indicating battery degradation. Conversely, "Max. Voltage Dischar. (V)" shows a strong positive correlation (0.78) with "RUL," indicating its potential as a predictor for the remaining life. Features like "Charging time (s)" and "Time constant current (s)" have strong positive correlations (above 0.90) with "Discharge Time (s)," highlighting interdependencies in the charging and discharging processes. Overall, the matrix underscores how specific battery parameters correlate with the RUL, providing insights for predictive maintenance and battery health monitoring.

4.3. Simulation Analysis of Proposed Scheme over Traditional Methods

In this section, the effectiveness achieved by the introduced method over the existing schemes is deliberated via graphical illustration. Several existing methods like LSTM-CNN, BiLSTM, GRU-RNN, and PSR-SVR techniques are compared with the proposed WOA_MTS-BiLSTM technique framework. The comprehensive analysis of the attained efficacy is depicted below.

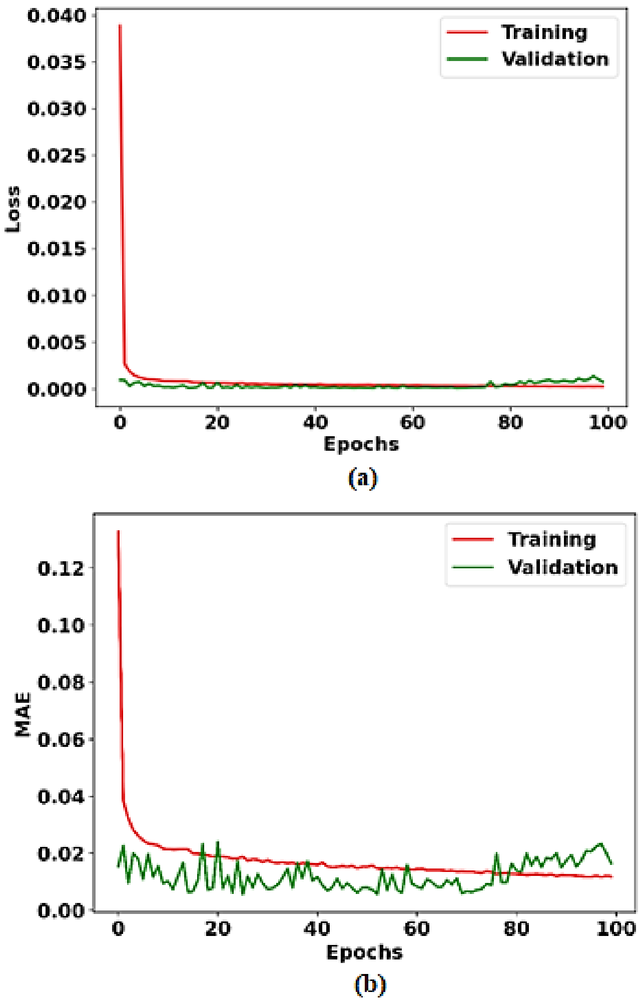

Figure 5. Training and Validation Analysis, (a) Loss and (b) MAE

Figures 5(a) and 5(b) indicate the training and validation analysis for loss and MAE, respectively. In *figure 5(a)*, the loss function, which measures the model's error during training, starts at approximately 0.04 for the training set and 0.03 for the validation set at epoch 1. As training progresses, the training loss decreases consistently, reaching approximately 0.002 at epoch 100. In *figure 5(b)*, initially, at epoch 1, the training MAE is approximately 0.12, while the validation MAE starts slightly lower, around 0.1. The training MAE steadily decreases over the epochs, reaching a minimum value near 0.002 at epoch 100. The validation MAE fluctuates in the earlier epochs but eventually aligns with the training MAE, stabilizing at approximately 0.002.

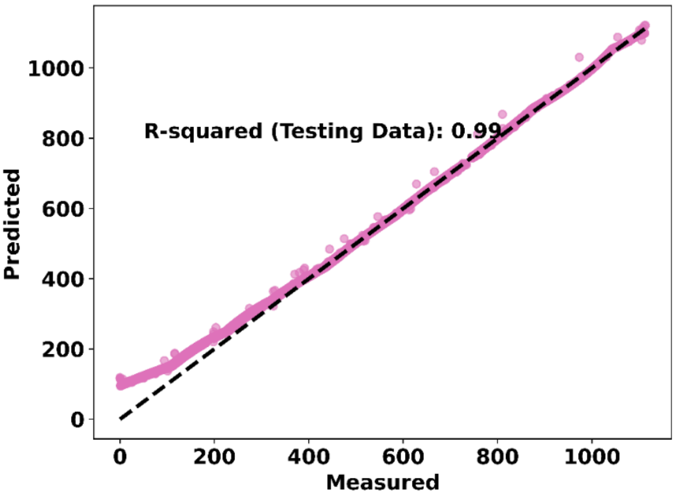

Figure 6. R-squared Analysis for the RUL Testing Data

Figure 6 illustrates the R-squared Analysis for the RUL testing data. The R-squared value is explicitly shown as 0.99, signifying that 99% of the variability in the measured data is explained by the predicted data. The diagonal line represents the ideal fit, and the plotted points closely align with this line, further emphasizing the model's high precision in forecasting RUL values. The high R² value and minimal deviation from the diagonal line indicate the model's effectiveness in seizing the underlying trends in the testing data.

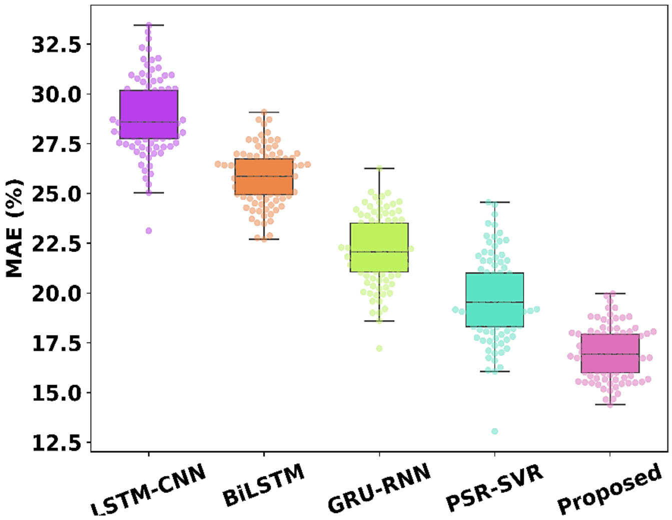

Figure 7. MAE Analysis for Varying Techniques

Figure 7 emphasizes the MAE Analysis for varying techniques. The MAE graph compares the MAE values for different models, including LSTM-CNN, BiLSTM, GRU-RNN, PSR-SVR, and the proposed WOA_MTS-BiLSTM model. The box plot shows that the proposed WOA_MTS-BiLSTM model achieves the lowest MAE, with a median value significantly below the other models, approximately at 12.5. The LSTM-CNN model demonstrates the highest MAE, with a median near 30, indicating lower prediction accuracy. The variability in error is also lower in the proposed WOA_MTS-BiLSTM model than in other models, showcasing its stability and reliability in predicting RUL. The GRU-RNN and PSR-SVR models perform moderately, with median MAE values around 22.5 and 17.5, respectively, while BiLSTM achieves a median close to 27.5. *Table 2* tabulates the experimental outcomes of the MAE metric.

Table 2. Experimental Outcomes of the MAE metric

Technique Used	Obtained Values
LSTM-CNN	28.7%
BiLSTM	25.8%
GRU-RNN	22.35%
PSR-SVR	19.42%
Proposed (WOA_MTS-BiLSTM)	**17.048%**

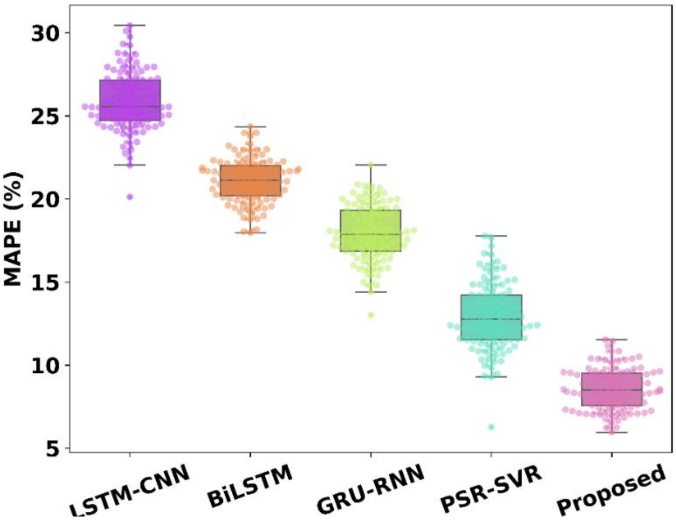

Figure 8. MAPE Analysis for Varying Techniques

Figure 8 emphasizes the MAPE Analysis for varying techniques. In the MAPE graph, the proposed WOA_MTS-BiLSTM model exhibits the smallest MAPE with a median value of approximately 5, highlighting its superior accuracy in percentage-based error metrics. In contrast, the LSTM-CNN model has the highest MAPE, with a median close to 30, reflecting a much larger deviation from the true RUL. The PSR-SVR model achieves a slightly better result than GRU-RNN, with median MAPE values around 10 and 15, respectively. BiLSTM, similar to its performance in MAE, shows a median MAPE of approximately 25, which indicates a consistent but less accurate trend compared to the proposed WOA_MTS-BiLSTM model. *Table 3* tabulates the experimental outcomes of the MAPE metric.

Table 3. Experimental Outcomes of the MAPE Metric

Technique Used	Obtained Values
LSTM-CNN	25.7%
BiLSTM	21.08%
GRU-RNN	18.15%
PSR-SVR	12.65%
Proposed (WOA_MTS-BiLSTM)	**8.63%**

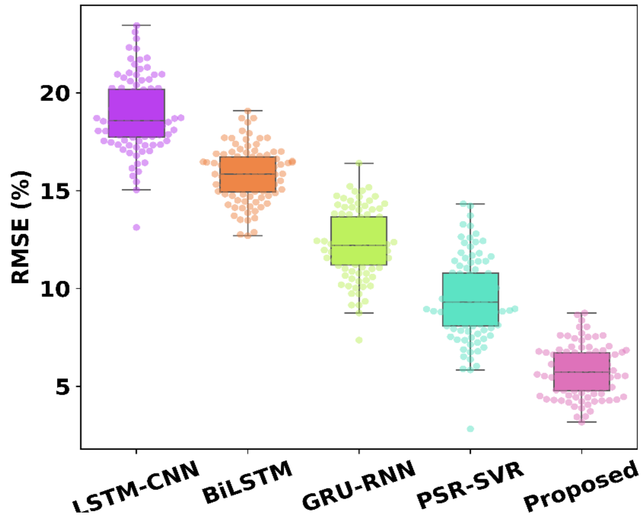

Figure 9. RMSE Analysis for Varying Techniques

Figure 9 emphasizes the RMSE Analysis for varying techniques. In the RMSE plot, the proposed WOA_MTS-BiLSTM model again outperforms others with a median RMSE of around 5%, indicating minimal deviation from the actual RUL values. The LSTM-CNN model records the largest RMSE, with a median value near 20%, showing poor performance in minimizing squared deviations. The BiLSTM model displays a higher RMSE compared to GRU-RNN and PSR-SVR, with median values around 17.5%, 12.5%, and 10%, respectively. The box plot for the proposed WOA_MTS-BiLSTM model reveals a tight distribution, demonstrating its robustness and reduced prediction error variance. *Table 4* tabulates the experimental outcomes of the RMSE metric.

Table 4. Experimental Outcomes of the RMSE metric

Technique Used	Obtained Values
LSTM-CNN	18.7%
BiLSTM	15.8%
GRU-RNN	12.5%
PSR-SVR	9.2%
Proposed (WOA_MTS-BiLSTM)	5.839%

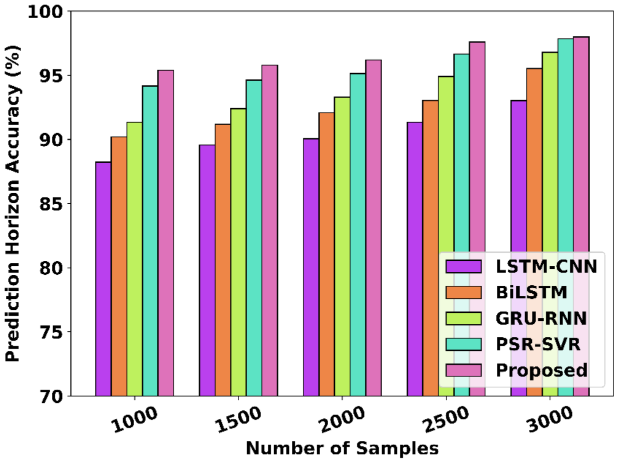

Figure 10. PHA Analysis for Varying Number of Samples

Figure 10 contemplates the PHA analysis for varying numbers of samples. From the graphical illustration, it is obvious that the proposed WOA_MTS-BiLSTM technique obtained better accuracy than conventional techniques. For 1000 samples, the existing LSTM-CNN, BiLSTM, GRU-RNN, PSR-SVR, and proposed WOA_MTS-BiLSTM technique obtained a PHA of 88.23%, 90.19%, 91.34%, 94.17%, and 95.4% respectively. For 2000 samples, the existing LSTM-CNN, BiLSTM, GRU-RNN, PSR-SVR, and proposed WOA_MTS-BiLSTM technique obtained a PHA of 90.04%, 92.08%, 93.29%, 95.13%, and 96.2% respectively. For 3000 samples, the existing LSTM-CNN, BiLSTM, GRU-RNN, PSR-SVR, and proposed WOA_MTS-BiLSTM technique obtained a PHA of 93.03%, 95.52%, 96.78%, 97.84%, 98% respectively. For 3000 samples, the existing LSTM-CNN, BiLSTM, GRU-RNN, PSR-SVR, and proposed WOA_MTS-BiLSTM technique obtained a PHA of 93.03%, 95.52%, 96.78%, 97.84%, 98% respectively. *Table 5* tabulates the experimental outcomes of the PHA metric.

Table 5. Experimental Outcomes of PHA metric

Technique Used	Number of Samples				
	1000	1500	2000	2500	3000
LSTM-CNN	88.231%	89.559%	90.044%	91.353%	93.034%
BiLSTM	90.197%	91.183%	92.081%	93.043%	95.522%
GRU-RNN	91.342%	92.406%	93.298%	94.907%	96.786%
PSR-SVR	94.173%	94.628%	95.139%	96.661%	97.843%
Proposed (WOA_MTS-BiLSTM)	95.4%	95.8%	96.2%	97.6%	98%

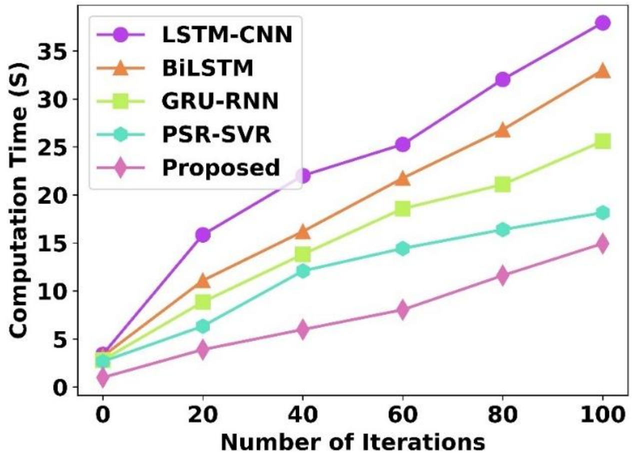

Figure 11. CT Analysis for Varying Iterations

Figure 11 depicts the CT analysis for varying iterations. The proposed WOA_MTS-BiLSTM model consistently demonstrates the lowest computation time, starting from approximately 2 seconds at 0 iterations and reaching around 15 seconds at 100 iterations. In contrast, other methods like LSTM-CNN and BiLSTM show significantly higher computation times, peaking at 35 seconds and 30 seconds, respectively, by 100 iterations. The lower computational overhead of the proposed WOA_MTS-BiLSTM method highlights its efficiency and suitability for real-time applications in RUL prediction. *Table 6* tabulates the experimental outcomes of the CT metric.

Table 6. Experimental Outcomes of CT Metric

Techniques used	Number of Iterations					
	0	20	40	60	80	100
LSTM-CNN	3.39s	15.8	22.0	25.2	32.0	37.9
BiLSTM	3.2s	11.0	16.1	21.7	26.8	32.9
GRU-RNN	2.8s	8.85	13.8	18.5	21.0	25.5
PSR-SVR	2.65	6.33	12.0	14.4	16.3	18.1
Proposed (WOA_MTS-BiLSTM)	**0.98**	**3.89**	**5.98**	**8.04**	**11.6**	**14.9**

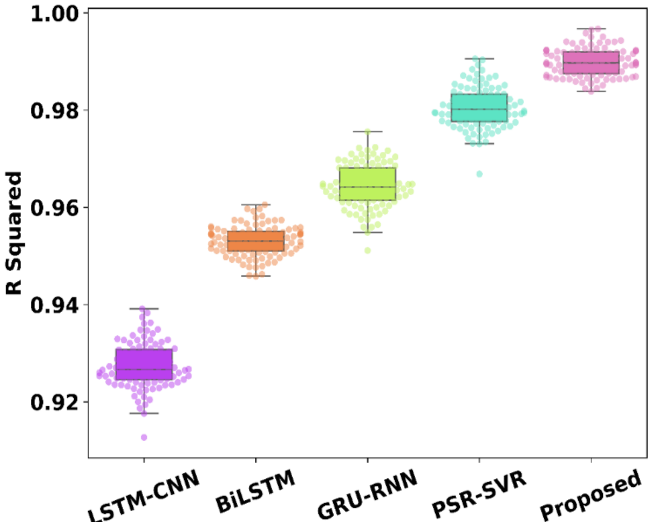

Figure 12. R-squared Analysis for Varying Techniques

Figure 12 contemplates the R-squared analysis for varying techniques. The R-squared metric measures the proportion of variance explained by the model, with a value closer to 1 indicating better performance. For the LSTM-CNN model, the R^2 values predominantly range between 0.92 and 0.94, reflecting lower predictive accuracy compared to other models. The BiLSTM model shows slightly better performance, with R-squared values centered between 0.94 and 0.96. The GRU-RNN model demonstrates further improvement, achieving R-squared values predominantly in the range of 0.96 to 0.975. The PSR-SVR model exhibits R-squared values slightly higher than GRU-RNN, primarily between 0.97 and 0.98, indicating an incremental enhancement in accuracy. The proposed WOA_MTS-BiLSTM model outperforms all others, with R-squared values clustering tightly between 0.98 and 1.00, reflecting its superior capability in explaining the variance in the data. This progression of R-squared values highlights the increasing effectiveness of the models, culminating in the proposed WOA_MTS-BiLSTM approach as the most accurate among those compared. *Table 7* tabulates the experimental outcomes of the R2 metric.

Table 7. Experimental Outcomes of R-squared Metric

Technique Used	Obtained Values
LSTM-CNN	0.927
BiLSTM	0.953
GRU-RNN	0.965
PSR-SVR	0.98
Proposed (WOA_MTS-BiLSTM)	**0.99**

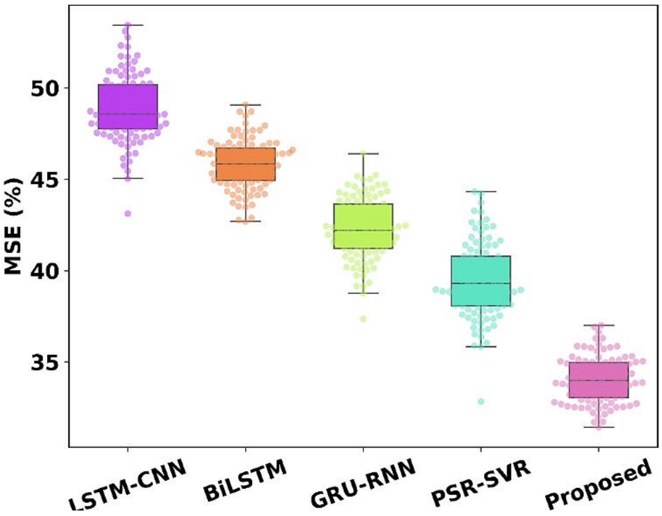

Figure 13. MSE Analysis for Varying Techniques

Figure 13 emphasizes the MSE Analysis for varying techniques. The MSE is a metric that evaluates the average squared difference between predicted and actual values, with lower values indicating better performance. For the LSTM-CNN model, the MSE values are concentrated around 50, signifying the lowest performance among the models. The BiLSTM model demonstrates an improvement, with MSE values clustering around 45. The GRU-RNN model further reduces the error, with MSE values centered near 42. The PSR-SVR model performs even better, achieving MSE values around 40. Finally, the proposed WOA_MTS-BiLSTM model exhibits the best performance, with MSE values tightly distributed around 35, indicating the smallest prediction errors among all models. This decreasing trend in MSE highlights the superior accuracy of the proposed WOA_MTS-BiLSTM approach compared to the alternatives. *Table 8* tabulates the experimental outcomes of the MSE metric.

Table 8. Experimental Outcomes of the MSE metric

Technique Used	Obtained Values
LSTM-CNN	48.7
BiLSTM	45.8
GRU-RNN	42.5
PSR-SVR	39.2
Proposed (WOA_MTS-BiLSTM)	34.096

4.4. Ablation Study Analysis

This section performs the ablation analysis for the developed scheme to determine the outcomes achieved over different modules. Four different modules are considered and various performance measures like MSE, RMSE, MAE, and MAPE are scrutinized. The investigation is made over without preprocessing (*i.e.*, without TG-AMF), without parameter tuning (*i.e.*, without WOA), without preprocessing, and without parameter tuning (*i.e.* TG-AMF + WOA), and the presence of all three stages (*i.e.,* presence of TG-AMF + MTS-BiLSTM + WOA. The brief analysis for each module and its obtained outcomes is blemished below;

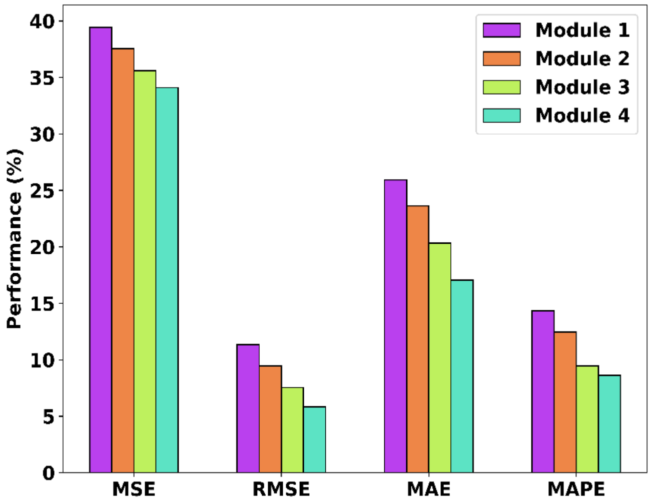

Figure 14. Ablation Study Analysis under Different Metrics

Figure 14 depicts the Ablation study Analysis under different Metrics. Module 4 indicates the presence of all three stages. For module 4, the MSE, RMSE, MAE, and MAPE achieved about 14.34, 12.45, 9.45, and 8.63, respectively. Module 3 indicates the absence of the TG-AMF + WOA. For module 3, the MSE, RMSE, MAE, and MAPE achieved about 25.93, 23.64, 20.34, and 17.04, respectively. Module 2 deliberates the absence of a parameter tuning process (i.e., without the WOA technique). For module 2, the MSE, RMSE, MAE, and MAPE achieved about 11.35, 9.45, 7.56, and 5.83, respectively. Module 1 encompasses the absence of the TG-AMF technique. For module 1, the MSE, RMSE, MAE, and MAPE achieved about 39.45, 37.56, 35.6, and 34.09, respectively. From the experimental outcomes, it is proven that the effective performance achieved with the presence of all three stages is defined by the proposed method. Table 9 tabulates the experimental outcomes of the ablation study plot for varying modules.

Table 9. Experimental Outcomes of Ablation Study Plot for Varying Modules

Modules	Assessment Measures			
	MSE	RMSE	MAE	MAPE
Without Preprocessing	39.45	11.35	25.93	14.34
Without parameter tuning	37.56	9.45	23.64	12.45
Without preprocessing and without parameter tuning	35.6	7.56	20.34	9.45
Presence of all three stages (TG-AMF + MTS-BiLSTM + WOA)	**34.09**	**5.839**	**17.04**	**8.630**

5. CONCLUSION

The proposed framework introduced and investigated an advanced approach to optimize the RUL forecast of EV Li-ion batteries, leveraging an MTS-BiLSTM network. By employing data from the RUL Prediction Dataset on Kaggle, the framework integrates a TG-AMF to enhance data quality by efficiently removing noise while preserving critical degradation features. The MTS-BiLSTM network effectively captures both Proximal correlations and Delayed dependencies in the data, ensuring accurate modeling of trends for reliable RUL forecasting. Further, the hyperparameters of the proposed model are fine-tuned using the WOA, significantly improving prediction accuracy while minimizing computational overhead. The robustness and effectiveness of the developed framework are demonstrated through benchmarking against state-of-the-art methods using key assessment measures, including R-squared, RMSE, MAE, MAPE, PHA, and CT. Notably, the proposed method achieves an overall PHA of 98%, MAE of 17.04, MAPE of 8.63, CT of 7.57 seconds, and RMSE of 5.83, underscoring its superior performance. However, several limitations remained in that the framework is validated only on a specific publicly available dataset, which may restrict its generalizability across different battery chemistries (e.g., LFP, NMC, LCO) and operational environments (e.g., temperature, charge/discharge rates). Second, while the model is optimized for computation, further improvements are needed to support real-time deployment in embedded or resource-constrained environments, such as on-board battery management systems (BMS) in electric vehicles. Third, although the proposed DL architecture exhibits strong predictive capabilities, interpretability and explainability remain limited. Future work should focus on evaluating the model across diverse datasets and battery types to ensure broader applicability. Techniques such as model pruning, quantization, or edge AI deployment strategies will also be explored to reduce latency and memory usage. Moreover, explainable AI (XAI) techniques (e.g., SHAP, LIME, attention mechanisms) will also be integrated to provide greater transparency into model decisions, enabling trust and insights for stakeholders such as engineers and battery manufacturers.

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

REFERENCES

[1] Wang, Shilong, Peiben Wang, Lingfeng Wang, Ke Li, Haiming Xie, and Fachao Jiang. "An enhanced deep learning framework for state of health and remaining useful life prediction of lithium-ion battery based on discharge fragments." Journal of Energy Storage 107 (2025): 114952.

[2] Reza, M.S., Hannan, M.A., Mansor, M., Ker, P.J., Rahman, S.A., Jang, G. and Mahlia, T.I., 2024. Towards enhanced remaining useful life prediction of lithium-ion batteries with uncertainty using an optimized deep learning algorithm. Journal of Energy Storage, 98, p.113056.

[3] Li, J., Zhao, S., Miah, M.S. and Niu, M., 2023. Remaining useful life prediction of lithium-ion batteries via an EIS-based deep learning approach. Energy Reports, 10, pp.3629-3638.

[4] CL, Sravanthi, Chandra Sekhar JN, N. Chinna Alluraiah, Dhanamjayulu C, Harish Kumar Pujari, and Baseem Khan. "An Overview of Remaining Useful Life Prediction of Battery Using Deep Learning and Ensemble Learning Algorithms on Data-Dependent Models." International Transactions on Electrical Energy Systems 2025, no. 1 (2025): 2242749.

[5] Wu, J., Kong, L., Cheng, Z., Yang, Y. and Zuo, H., 2022. RUL prediction for lithium batteries using a novel ensemble learning method. Energy Reports, 8, pp.313-326.

[6] Khan, M.K., Abou Houran, M., Kauhaniemi, K., Zafar, M.H., Mansoor, M. and Rashid, S., 2024. Efficient state of charge estimation of lithium-ion batteries in electric vehicles using evolutionary intelligence-assisted GLA–CNN–Bi-LSTM deep learning model. Heliyon, 10(15).

[7] Li, Yuanjiang, Liping Li, Lei Li, Xinyu Huang, Guodong Sun, Yina Wang, and Jinglin Zhang. "Research on hybrid data-driven method for predicting the remaining useful life of lithium-ion batteries." Computer Physics Communications 309 (2025): 109500.

[8] Bak, Gwiman, and Youngchul Bae. "Deep learning network for predicting the remaining useful life of lithium-ion batteries using the positive and negative convolution perceptron model." Journal of the Chinese Institute of Engineers 48, no. 4 (2025): 467-479.

[9] Hussien, S.A., BaQais, A. and Al-Gabalawy, M., 2024. Estimation of the residual useful life of EV batteries using advanced hybrid learning tools. Electrical Engineering, 106(3), pp.2651-2677.

[10] Zhao, S., Zhang, C. and Wang, Y., 2022. Lithium-ion battery capacity and remaining useful life prediction using board learning system and long short-term memory neural network. Journal of Energy Storage, 52, p.104901.

[11] Liu, Wei, and Jiashen Teh. "Remaining useful life prediction of lithium-ion batteries based on an incremental internal resistance aging model and a gated recurrent unit neural network." Energy (2025): 137527.

[12] Rastegarpanah, A., Asif, M.E. and Stolkin, R., 2024. Hybrid Neural Networks for Enhanced Predictions of Remaining Useful Life in Lithium-Ion Batteries. Batteries, 10(3), p.106.

[13] Granado, L., Ben-Marzouk, M., Saenz, E.S., Boukal, Y. and Jugé, S., 2022. Machine learning predictions of lithium-ion battery state-of-health for eVTOL applications. Journal of Power Sources, 548, p.232051.

[14] Lee, J., Sun, H., Liu, Y. and Li, X., 2024. A machine learning framework for remaining useful lifetime prediction of li-ion batteries using diverse neural networks. Energy and AI, 15, p.100319.

[15] Feng, Weiguo, Zhongtian Sun, Yilin Han, Nian Cai, and Yinghong Zhou. "A multi-strategy attention regression network for joint prediction of state of health and remaining useful life of lithium-ion batteries using only charging data." Journal of Power Sources 636 (2025): 236507.

[16] Li, X., Yu, D., Byg, V.S. and Ioan, S.D., 2023. The development of machine learning-based remaining useful life prediction for lithium-ion batteries. Journal of Energy Chemistry, 82, pp.103-121.

[17] Wang, S., Ma, H., Zhang, Y., Li, S. and He, W., 2023. Remaining useful life prediction method of lithium-ion batteries is based on variational modal decomposition and deep learning integrated approach. Energy, 282, p.128984.

[18] Zraibi, Brahim, Mohamed Mansouri, and Salah Eddine Loukili. "RUL prediction for lithium-ion batteries using improved-CGD hybrid model." International Journal of Dynamics and Control 13, no. 6 (2025): 237.

[19] Shi, J., Rivera, A. and Wu, D., 2022. Battery health management using physics-informed machine learning: Online degradation modeling and remaining useful life prediction. Mechanical Systems and Signal Processing, 179, p.109347.

[20] Zhou, Y., Wang, S., Xie, Y., Shen, X. and Fernandez, C., 2023. Remaining useful life prediction and state of health diagnosis for lithium-ion batteries based on improved grey wolf optimization algorithm-deep extreme learning machine algorithm. Energy, 285, p.128761.

[21] Swain, D., Kumar, M., Nour, A., Patel, K., Bhatt, A., Acharya, B. and Bostani, A., 2024. Remaining Useful Life Predictor for EV Batteries Using Machine Learning. IEEE Access.

[22] Gao, D., Liu, X., Zhu, Z. and Yang, Q., 2023. A hybrid cnn-bilstm approach for remaining useful life prediction of evs lithium-ion battery. Measurement and Control, 56(1-2), pp.371-383.

[23] Mishra, S., Choubey, A., Reddy, B.A. and Misra, R., 2024. Enhancing EV lithium-ion battery management: automated machine learning for early remaining useful life prediction with innovative multi-health indicators. The Journal of Supercomputing, pp.1-48.

[24] Zhao, J., Ling, H., Liu, J., Wang, J., Burke, A.F. and Lian, Y., 2023. Machine learning for predicting battery capacity for electric vehicles. ETransportation, 15, p.100214.

[25] Cai, S., Hu, J., Ma, S., Yang, Z. and Wu, H., 2022. Remaining useful life prediction method of EV power battery for DC fast charging condition. Energy Reports, 8, pp.1003-1010.

[26] Jafari, S., Shahbazi, Z. and Byun, Y.C., 2022. Lithium-ion battery health prediction on hybrid vehicles using machine learning approach. Energies, 15(13), p.4753.

[27] Ma, Liang, Jinpeng Tian, Tieling Zhang, Qinghua Guo, and Chi Yung Chung. "Enhanced battery life prediction with reduced data demand via semi-supervised representation learning." Journal of Energy Chemistry 101 (2025): 524-534.

[28] https://www.kaggle.com/code/sasakitetsuya/remaining-useful-lifetime-prediction/notebook

[29] Fatima, S.H., Munir, A. and Hussain, S.T., 2024. Image denoising using difference classifier and trimmed global mean filter adaptive approach. The Visual Computer, 40(8), pp.5309-5321.

[30] Zou, Q., Xiong, Q., Li, Q., Yi, H., Yu, Y. and Wu, C., 2020. A water quality prediction method based on the multi-time scale bidirectional long short-term memory network. Environmental Science and Pollution Research, 27, pp.16853-16864.

[31] Han, M., Du, Z., Yuen, K.F., Zhu, H., Li, Y. and Yuan, Q., 2024. Walrus optimizer: A novel nature-inspired metaheuristic algorithm. Expert Systems with Applications, 239, p.122413.