IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 22 July 2023, accepted 6 August 2023, date of publication 14 August 2023, date of current version 18 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3304699

RESEARCH ARTICLE

Optimizing Battery RUL Prediction of Lithium-Ion Batteries Based on Harris Hawk Optimization Approach Using Random Forest and LightGBM

SADIQA JAFARI[1] AND YUNG-CHEOL BYUN[2]

[1]Department of Electronic Engineering, Jeju National University, Jeju 63243, South Korea
[2]Department of Computer Engineering, Major of Electronic Engineering, Institute of Information Science and Technology, Jeju National University, Jeju 63243, South Korea

Corresponding author: Yung-Cheol Byun (ycb@jejunu.ac.kr)

This research was financially supported by the Ministry of Small and Medium-sized Enterprises (SMEs) and Startups (MSS), Korea, under the "Regional Specialized Industry Development Plus Program (R&D, S3246057)" supervised by the Korea Technology and Information Promotion Agency for SMEs (TIPA).

ABSTRACT Predictive Maintenance (PdM) of lithium-ion batteries has garnered significant attention in recent years due to their widespread application as energy supplies in various industrial equipment, including automated guided vehicles and battery Electric Vehicles (EVs). Accurately estimating these batteries' Remaining Useful Life (RUL) is crucial for ensuring optimal performance, preempting unexpected failures, and minimizing maintenance costs. This article focuses on the importance of RUL prediction for lithium-ion batteries and its implications in predictive maintenance. We suggest a novel method based on machine learning techniques using optimization parameters to accurately predict the RUL of lithium-ion batteries. Our approach uses several battery performance variables, such as voltage, current, and temperature, to build a prediction model to anticipate the battery's RUL precisely. We compare the performance of our suggested process with existing models for battery RUL prediction, incorporating Harris Hawks Optimization (HHO) for hyperparameter tuning. We evaluate the performance of our approach on a dataset of lithium-ion batteries and compare it with other related methods. On a dataset of lithium-ion batteries, we assess our method's effectiveness and contrast it with other relevant techniques. The proposed method achieves high accuracy in predicting RUL, as evidenced by low values of metrics such as MAE, MSE, RMSE, MAPE, R^2, and NMRSE. Also, it achieves high R^2 scores of 0.979 and 0.971 for the training and testing data, suggesting the model's high effectiveness in predicting the RUL of batteries.

INDEX TERMS RUL prediction, random forest, LightGBM, Harris Hawk optimization, predictive maintenance, lithium-ion batteries.

I. INTRODUCTION

Society is pressured to research and produce new energy due to the traditional energy dilemma. Lithium-ion batteries are among the most frequently used new energy sources due to their high energy density, high output voltage, prolonged cycle life, and wide operating temperature range [1]. But as lithium-ion batteries are repeatedly charged and drained, the internal resistance increases. The battery warms up significantly once the internal resistance increases, impacting the battery pack's functionality and everyday employ [2]. Routine and experience-based inspection and maintenance have been widely used as preventative maintenance techniques for many years. However, a more reliable and mature digital system called the Battery Management System (BMS) has been developed with the digital transformation trend and advancements in lithium-ion battery technology. The BMS can use the Internet of Things (IoT) to gather big data from numerous large-scale sensors and track the status of lithium-ion batteries in real time. A dependable and intelligent BMS uses

The associate editor coordinating the review of this manuscript and approving it for publication was Branislav Hredzak.

precise sensor data analytics to prevent abrupt device failures and save significant maintenance costs while providing early warning signals of future degradation. Artificial intelligence approaches are utilized to create efficient prediction models that correctly forecast the RUL and enable early defect detection of lithium-ion batteries to achieve a smart BMS [3]. Conventional RUL prediction typically uses the data from the acceleration sensor. This approach can produce accurate RUL predictions if it exclusively targets mechanical machinery like bearings [4].

PdM has evolved an essential strategy for improving the reliability and efficiency of industrial systems. By operating sensor data to detect potential equipment failures before they occur, organizations can reduce downtime and maintenance expenses and raise the safety and productivity of their processes. In recent years, ML algorithms have shown significant promise in PdM. RF and LightGBM are two famous algorithms widely operated in RUL prediction for various kinds of equipment. ML algorithms have the advantage of being able to process vast volumes of sensor data and extract complicated patterns that are challenging for human specialists to recognize. The choice of hyperparameters, which are parameters that regulate the algorithm's learning process, has a significant impact on the performance of ML models. In this paper, we utilize the harmony search algorithm to optimize the hyperparameters of the RF and LightGBM models for predicting the RUL of lithium-ion batteries. We analyze the impact of hyperparameters on the accuracy of RUL predictions and compare the performances of the two algorithms. The findings demonstrate that sensor data integration and hyperparameter optimization may significantly improve lithium-ion battery RUL predictions. Our study contributes to PdM by providing valuable insights into utilizing ML algorithms and sensor data for RUL prediction, which can help reduce maintenance costs, prevent equipment failures, and improve safety across various applications.

Our study aims to contribute to the field; the following list summarizes our contributions:

- Developing a technique for predicting the RUL of NMC-LCO 18650 batteries employing ML approaches.
- Comparing the performance of the suggested approach with different models, the related work for battery RUL prediction.
- Using HHO for hyperparameter tuning to optimize the performance of the ML models.
- Highlighting the potential of ML methods for battery RUL prediction, which can contribute to the field of PdM.
- Demonstrating the effectiveness of the proposed method in predicting the RUL of lithium-ion batteries, as evidenced by high R^2 scores and low values of metrics such as MAE, MSE, RMSE, MAPE, and NMRSE.
- Comparing the performance of two ML algorithms, RF and LightGBM, for battery RUL prediction.

The remainder of the paper is structured as follows: In Section II, we briefly review the relevant literature on PdM and battery health monitoring. Section III present our proposed method, which combines RF and LightGBM with HHO for hyperparameter tuning. We also discuss the use of data curation and analysis in Section IV. In Section V, we present the implementation process, results, and performance evaluation of our method. Finally, we conclude the paper in Section VI, where we summarize our contributions and highlight future research directions.

II. LITERATURE REVIEW

This section presents related work of PdM and the RUL using ML. Estimating the remaining usable life is a critical PdM task that has grown important as a study area during the past ten years. In order to build a successful maintenance strategy, RUL prediction is essential to ensuring the overall system's dependability and safety. Data from the past can be used to forecast RUL. The RUL of assets may be predicted with increasing accuracy using ML approaches. PdM is condition-based maintenance that involves routinely inspecting the equipment. PdM's ultimate objective is to predict when an appliance will develop defects to avoid unanticipated equipment breakdowns and, as a result, save maintenance costs and downtime. Several alternative ML approaches have been developed in order to predict the matching RUL using the gathered historical data [5]. Different approaches have been utilized in previous studies, including classification, clustering, and regression techniques, depending on the nature of the issue. For instance, some studies have concentrated on classifying whether a failure would occur, while others have employed clustering approaches to group similar events. In addition to classification techniques, regression methods have also been utilized to predict numerical target variables, as demonstrated by Uhlmann et al. [6]. Ellefsen et al. [7] investigated unsupervised learning for predicting RUL using a semi-supervised process. Other researchers have explored the potential of deep learning strategies, such as Convolutional Neural Networks (CNN), Deep Neural Networks (DNN), and Long Short-Term Memory (LSTM), for estimating RUL [8], [9], [10].

A. RUL BASED ON THE MACHINE LEARNING METHODS

The primary difficulties and future directions for study in battery health prognostics are outlined in this paper's thorough examination of aging processes and cutting-edge health prognostic techniques. There has been a review and description of the complicated connections between aging processes, aging modes, influencing elements, and aging kinds. After a comprehensive review of each prognostic job and method, the battery health prognosis approaches are separated into groups depending on various time scales and purposes, such as short-term SOH estimation, long-term End Of Life (EOL) prediction, and deterioration trajectory prediction. Comparative analyses have been conducted after the study has explored the similarities and differences between the various RUL battery-related procedures [11]. The combination of the enhanced variational modal decomposition (VMD),

particle filter (PF), and Gaussian Process Regression (GPR) in this study has been recommended as a unique hybrid technique for estimating battery future capacity and RUL, which the recorded battery capacity data is divided by the VMD algorithm into several residual sequences and an aging trend sequence, with the number of modal layers being determined then the aging trend sequence and residual sequences are then, respectively, have been predicted with using the prediction models of the PF and GPR algorithms [12]. These days for the RUL issue in the battery, there are many techniques that many researchers use, such as deep learning-based forecasting techniques, in an attempt to enhance predicted outcomes; some current deep learning-based forecasting techniques focus on increasing the network depth and expanding the training dataset, which can be highly resource-intensive. The research has suggested a merging technique based on the Long Short-Term Memory Neural Network (LSTM NN) and the Broad Learning System (BLS) algorithm. A fusion neural network model has also been developed to predict the lithium-ion battery capacity and RUL correctly. To be more precise, the BLS algorithm creates feature nodes initially using historical capacity data and then creates enhancement nodes by using enhancement mapping. The input layer of the LSTM NN is then formed by concatenating all BLS-created nodes to build the BLS-LSTM fusion NN [13].

After the IC curves are generated using the voltage data from the constant current charging phase, they are denoised using the smoothing spline filter. Using the Pearson correlation coefficient technique, the significant HIs have been selected based on the IC curve properties. The L2 regularization parameter formed the BLS network, which was then tuned to create the SOH estimate model. Based on the Incremental Capacity Analysis (ICA) and enhanced BLS network-based SOH estimate technology for lithium-ion batteries, a PSO approach has filtered the shrinkage scale of the enhancement nodes [14]. In order to predict battery lifespan, this research has suggested a revolutionary semi-supervised self-learning method in which the partial capacity-voltage curve has been employed to obtain three Health Indicators (HIs). Information from three randomly picked batteries in the original part has been utilized to create the capacity estimate and lifespan prediction models. The historical capabilities have been recreated operating the HIs to offer fictitious values for the lifetime model's self-training. In the research, future deterioration has been predicted utilizing the self-trained lifespan model to provide a probabilistic forecast of future capacity [15]. The novel cell-to-pack health and lifespan prognostics method presented in the paper is based on combining transferable deep learning and GPR, where the partial discharge operation generates general health markers. The sequential degradation model of the HIs has been developed and moved based on a deep learning architecture for the prediction of battery pack deterioration; following that, the probabilistically anticipated future deteriorated capacities of the battery pack and each battery cell have provided a complete lifespan prognosis. Additionally, experimental results have been shown that define the weakest cell for maintenance in advance and the battery pack's desired capacity movements [16]. A new approach for estimating RUL has been suggested and is based on live model modification employing transfer learning and optimal HIs. GPR has been utilized to optimize the threshold for HIs to identify the EOL. The RUL has been predicted based on the optimized HIs operating a mix of transfer learning and gated RNNs, which may support online applications. The prediction model is further fine-tuned to provide accuracy in performing the test batteries before the deterioration cycle data. The model was trained initially employing a suitable battery [17].

The study aims to enhance the accuracy of RUL prediction for lithium-ion batteries, which is crucial for their prognostics and health management. The authors proposed a two-phase deterioration model with a dynamic transition point utilizing binary segmentation. This model considers the various ways that tested battery cells degrade, resulting in more precise RUL predictions. The authors created a system based on particle filtering to account for uncertainties during RUL prediction [18]. The research has recommended a specific RUL prediction technique for lithium-ion batteries based on an LSTM network designed for use with an Improved Sparrow Search Algorithm (ISSA) for lithium-ion batteries. Because they directly affect prediction accuracy, the LSTM hyperparameters that needed tuning were chosen. ISSA then adjusts the LSTM's hyperparameters depending on data about battery capacity from several datasets to produce RUL prediction [19]. Accurate RUL prediction is necessary for supercapacitors to operate securely and reliably. The Bidirectional Long (BiLSTM) network and the observer are the foundations of the two-stage online RUL prediction framework that has been proposed in this study. The Bayesian Optimization (BO) technique and the BiLSTM network have been employed in stage 1 to estimate capacitance. The BiLSTM network's hyperparameters have been intended to be optimized by the BO method. The paper has used short-term variations to reduce the Moving Average Filter (MAF) employed in Stage 2 to manage the estimated capacitance. The double exponential model has represented the deterioration trajectory, which has been updated by the observer using the estimated capacitance as the measurement [20].

The proposed strategy propagates these points utilizing the unscented KF to produce a complex proposal distribution, then employs singular value decomposition to provide sigma samples for the unscented transformation. Compared to the traditional unscented particle filter method, the improved method offers better performance in proposing the RUL of storage batteries [21]. A hybrid framework that integrates model-based and data-driven techniques has been proposed to enhance the accuracy of battery lifetime prediction, which leverages an empirical model based on physical principles and a support vector regression model based on actual operational data to estimate the battery lifetime by computing the battery capacity. In order to ensure that the model parameters

remain current and accurate, particle filters are employed to update the practical model dynamically in [22]. Reference [23] has demonstrated how to forecast how much useable life a battery has left using a unique hybrid Elman-LSTM technique that combines LSTM and ENN with the experimental model decomposition methodology. The empirical mode decomposition method divides the recorded battery capacity against cycle number data into various sub-layers. The high-frequency and low-frequency sub-layers are then predicted using recurrent LSTM and Elman-NN models. In Table 1, there is a comparison of several model-based methods for RUL prediction.

III. MATERIALS AND METHODS

This section provides a thorough discussion of the suggested model.

A. PROPOSED FRAMEWORK

The strategy's main goal is to create a PdM model that reliably predicts the RUL of lithium-ion batteries using sensor data. The three primary parts of the approach are the preparation and gathering of data, the creation of the model, and the assessment of the model. The method's main goal is to create a PdM model that reliably predicts the RUL of lithium-ion batteries employing sensor data. The methodology comprises three main components: data collection and preparation, model development, and model evaluation. The required sensor data is collected and pre-processed in the first step to remove any noise. The pre-processed data is then used to extract relevant features using statistical and machine-learning techniques. In the second step, two ML algorithms, namely RF and LightGBM, are employed to develop the predictive model. Both algorithms are trained and optimized using hyperparameters to improve their performance.

Further optimization of the model's parameters using HHO is also used to get a more precise forecast of the RUL. In the final step, the developed models are evaluated by operating various metrics. Based on their sensor data, the best-performing model is then selected and utilized to predict the RUL of new lithium-ion batteries. The methodology generally involves a comprehensive approach that includes data preparation, feature extraction, model development, hyperparameter optimization, and model evaluation, ultimately leading to an accurate and reliable RUL predictive model for Lithium-Ion batteries. Figure 1 depicts the suggested approach's implementation flowchart.

B. COMPUTATIONAL EFFORT AND COMPLEXITY

The computational effort and complexity in the RF and LightGBM models, using HHO for RUL prediction in batteries, refer to the computational resources needed for hyperparameter tuning and model training. During hyperparameter tuning with HHO, the algorithm explores the hyperparameter space to find the optimal settings for each model. This process involves a certain number of optimization iterations and a specified population size, impacting the overall computational effort. After hyperparameter tuning, the optimized models are trained. Model training entails constructing decision trees RF or applying gradient boosting (LightGBM), adding to the computational complexity. The dataset size, number of trees/boosting rounds, and other model-specific settings further influence the computational effort during training. To assess computational efficiency, we compare the time required for hyperparameter tuning and model training between the RF and LightGBM models. LightGBM typically exhibits higher complexity due to its gradient-boosting technique. Evaluating the computational effort and complexity offers insights into the feasibility and efficiency of using RF and LightGBM with HHO for battery RUL prediction.

C. HARRIS HAWKS OPTIMIZATION

HHO is a meta-heuristic optimization technique inspired by Harris hawks' hunting methods. The algorithm mimics the strategy these birds use to catch prey, which involves cooperation and competition among members of the hawk group. HHO begins with an initial population of hawks, where each hawk represents a candidate solution. The algorithm then proceeds through a series of iterations, or generations, during which the hawks search for the optimal solution. In each generation, the hawks are evaluated based on their fitness, which is determined by a fitness function that measures the quality of the candidate solution. The HHO algorithm has several key features that make it particularly well-suited for optimization problems in engineering and other fields [29]. Figure 2 represents the system architecture of HHO.

First, the algorithm is easy to implement and requires minimal tuning of its parameters. Second, a global optimization algorithm can search the entire solution space for the best possible solution. Finally, the algorithm is relatively fast, which makes it particularly useful for large-scale optimization problems. HHO is a useful tool for optimizing the machine-learning algorithm parameters for PdM and RUL prediction in the context of lithium-ion batteries. Through parameter optimization, HHO can significantly enhance the accuracy and reliability of RUL prediction, resulting in more efficient maintenance planning and cost savings for battery manufacturers and users.

$$x_{i,j}(t+1) = x_{i,j}(t) + r_1 \left(x_{best,j}(t) - x_{i,j}(t)\right) r_2 \left(x_{p,j}(t) - x_{q,j}(t)\right) \tag{1}$$

where $x_{i,j}(t)$ is the j-th component of the i-th hawk's position at time t, $x_{best,j}(t)$ is the best position found so far in the search space, $x_{p,j}(t)$ and $x_{q,j}(t)$ are the positions of two randomly selected hawks, and r_1 and r_2 are random coefficients between 0 and 1. The position update equation is used to move each hawk to a new position in the search space based on the positions of other hawks and the best position found so far. HHO is a good optimization algorithm with excellent performance in many optimization problems. Its ability to balance cooperation and competition among population members and its simplicity and efficiency make

TABLE 1. Comparative Analysis of several model-based strategies for lithium-ion battery RUL prediction.

Approach	Method	Benefit	Disadvantage	Model results
[24]	1D Continuum model of electrochemistry	Presents a powerful basic model for lithium and sulfur research in the future.	Model description including SEI production and nucleation is incomplete.	Simulation results based on various C-rates, polysulfide the effect, shuttle effect, and endless charging is delivered.
[25]	Enhanced mutated PF technique	Calculations were made for the state estimate and RUL prediction accuracy.	The suggested model's error metrics were high.	Calculations were made for the state estimate and RUL prediction accuracy.
[26]	Kernel adaptive filter	Reduced complexity, improved robustness	Complexity in time and space is demonstrated.	Battery RUL prediction
[27]	Exponential model and PF	High predictability and convergence.	At initial start cycles, the prediction accuracy is poor owing to improper parameter selection.	Battery RUL prediction
[28]	Fractional order equivalent circuit model	Reconstruction of the EIS profile with accuracy.	Due to the omission of crucial elements like discharge rates and a changeable temperature profile, the prediction's accuracy is limited.	Battery RUL prediction

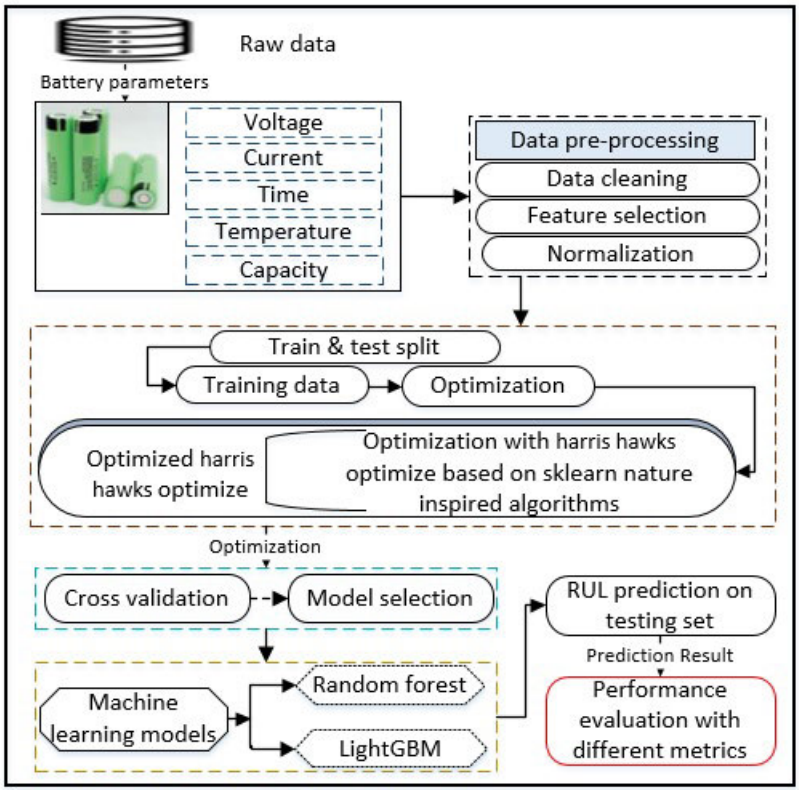

FIGURE 1. The overall framework for RUL prediction of lithium-ion battery.

it a valuable tool for optimization problems in engineering and other fields [30].

D. RANDOM FOREST ALGORITHM

In order to increase the precision and stability of predictions, RF is a well-liked ensemble learning approach in ML. Both classification and regression tasks can be accomplished using the supervised learning algorithm. Building numerous decision trees, each learned on a random subset of the training data and features, is the main goal of RF. A majority vote on each individual tree prediction during prediction determines the final result. One of the main advantages of RF is that it is relatively insensitive to the choice of hyperparameters and can handle a wide range of data types and distributions. It is a reliable approach for predictive modeling since it can deal with missing values and noisy data. Additionally, RF offers helpful insights into the relative weights of each feature in the prediction, which can aid in understanding the underlying data patterns. Figure 3 displays the RF algorithm's flow diagram.

We have a training set of N samples with p input features and an associated vector of N response values. To build a

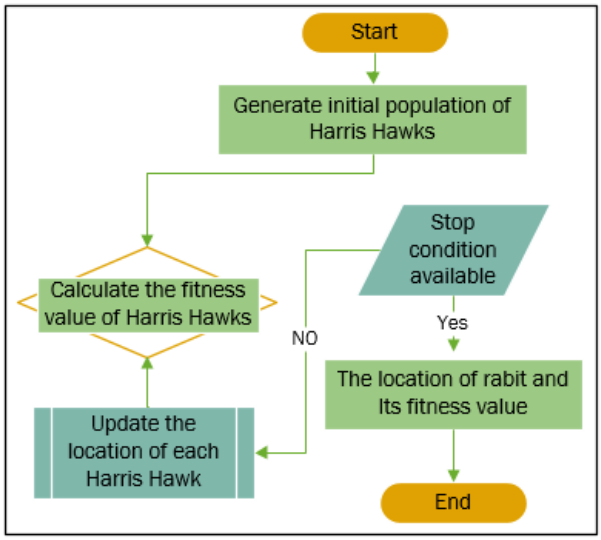

FIGURE 2. Flowchart of HHO.

decision tree that can accurately predict the response value for new samples. RF is an ensemble method that creates numerous decision trees and produces the average prediction of each tree's predictions. A subset of the training data (with replacement) and a subset of the input features at each split are required to build each decision tree in the randomly chosen RF. This randomness helps to prevent overfitting to the training data and improves the generalization performance on new data. We select the feature that provides the best split at each decision tree node according to a splitting criterion. The tree recursively develops by dividing the data into smaller groups according to the chosen feature and split point. Based on the sample's feature values, we move from the root node to a leaf node of the decision tree to forecast a new sample. The prediction is then made using the response value connected to the leaf node. Equation 2 represents the ensemble prediction of the RF algorithm [31].

$$\hat{y}(x, \Theta) = \frac{1}{M} \sum_{m=1}^{M} f(x; \theta_m) \tag{2}$$

where $\hat{y}$ is the predicted response value, $\mathbf{x}$ is the input vector, $\Theta = \{\theta_1, \ldots, \theta_M\}$ is the set of decision trees, and $f(x; \theta_m)$ is the output of the m-th decision tree.

The hyperparameters for the RF algorithm are:

- $n_{estimators}$: the forest's density of decision trees.
- $max_{features}$: the number of features to consider when looking for the best split at each node.
- max_{depth}: the maximum depth of the decision trees.
- $min_{samplessplit}$: the least amount of samples necessary to split an internal node.
- $min_{samplesleaf}$: the bare minimum of samples needed at a leaf node.

To optimize the RF model, we use HHO for hyperparameter tuning. HHO efficiently explores the hyperparameter space, maximizing the R^2 score during cross-validation. The comprehensive hyperparameter grid ($param_{grid}$) includes

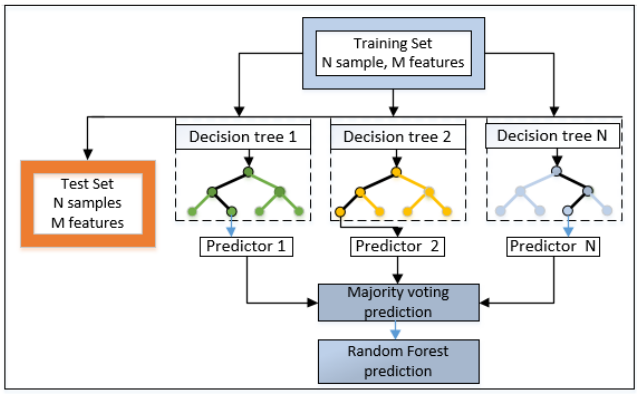

FIGURE 3. Flow diagram for RF algorithm structure.

$n_{estimators}$, $max_{features}$, max_{depth}, $min_{samplessplit}$, $min_{samplesleaf}$, and criterion. The NatureInspiredSearchCV class is instrumental in performing the hyperparameter optimization. This class orchestrates the optimization process by integrating the RF model, the hyperparameter grid, and the HHO algorithm. We specify important settings such as the population size, maximum number of generations, and number of runs, which influence the efficiency and effectiveness of the HHO algorithm. The RF model captures underlying patterns in the data, leading to improved RUL predictions. We validate the model's performance on diverse battery types, ensuring practical applicability.

E. LightGBM

The LightGBM gradient-boosting framework creates the usage of tree-based learning strategies. Similar to RF, it is also an ensemble method that builds multiple trees and combines their results. However, LightGBM uses a different approach to construct decision trees and train the model, which leads to faster and more accurate results. The main feature of LightGBM is the ability to handle large-scale data efficiently. It uses Gradient-based One-Side Sampling (GOSS) to select only a subset of the training data for each iteration, reducing computation time and memory usage. It also supports parallel and distributed computing, making it suitable for big data problems. The number of trees, learning rate, and the maximum depth of each tree must all be specified as hyperparameters before we can train a LightGBM model. LightGBM also offers several advanced features, such as definite feature handling and early stopping, which can improve the model's performance and reduce overfitting [32]. The objective function of LightGBM can be written as in Equation 3

$$Obj(\Theta) = \sum_{i=1}^{N} l(y_i, \hat{y}i) + \sum k = 1^K \Omega(f_k) \tag{3}$$

Given a dataset with N samples, where X_{ij} represents the value of the $j-th$ feature for the $i-th$ sample, and y_i is the corresponding label. Where l represents the loss function, $\hat{y}_i$ is the predicted value for the $i-th$ sample, Θ represents the set of all model parameters, including the decision trees and split

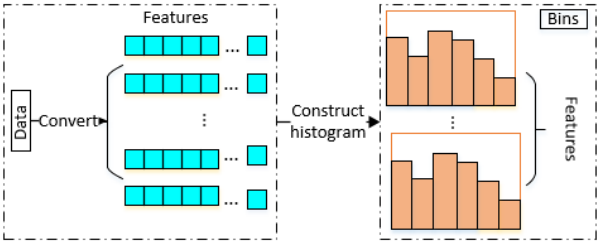

FIGURE 4. Diagram of the histogram optimization approach.

points, f_k is the $k-th$ decision tree, K is the total number of trees, and $\Omega(f_k)$ is the regularization term that penalizes the complexity of the model. Training LightGBM aims to find the set of model parameters Θ that minimizes the objective function $Obj(\Theta)$. It is typically done using a gradient-based optimization algorithm such as Stochastic Gradient Descent (SGD).

F. HISTOGRAM OPTIMIZATION STRATEGY

The Histogram Optimization Strategy is a feature of LightGBM that optimizes the creation and usage of histograms for faster training and better accuracy. Instead of predefining the histogram bins, LightGBM applies a dynamic binning approach that adapts to the data distribution. The algorithm starts by finding each feature's minimum and maximum values in the training data. Then, it divides each feature range into equal-width bins and constructs histograms based on the counts of samples falling into each bin. However, instead of using all the bins, LightGBM applies an optimization technique to select only the most informative ones. The optimization is performed by iteratively merging adjacent bins with the least gain in the loss function until the desired number of bins is reached. The Histogram Optimization Strategy has several advantages over other binning methods. First, it reduces memory consumption by using a smaller number of bins. Second, it enhances the model's accuracy by effectively capturing the data distribution. Third, it speeds up the training process by reducing the number of splits and simplifying the optimization problem. The flow diagram of the Histogram optimization strategy algorithm is shown in Figure 4.

In the histogram optimization strategy, LightGBM constructs histograms for each feature on the fly during the tree-building process. By utilizing this histogram-based computation, LightGBM can achieve faster and more efficient tree building than traditional decision tree algorithms. The objective function for finding the best split in LightGBM using the Histogram optimization strategy is Equation 4:

$$\Delta = \frac{1}{2}\left[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{(G_L+G_R)^2}{H_L+H_R+\lambda}\right] - \gamma \quad (4)$$

In Equation 4, the G_L and G_R are the gradient statistics for the potential split's left and right nodes. Where H_L and H_R are the sum of Hessians for the left and right nodes, respectively, and λ and γ are regularization parameters. The split with the most significant value of Δ is chosen as the best split.

IV. DATA CURATION AND ANALYSIS

This section provides a detailed overview of this study's data curation and analysis process using the NMC-LCO 18650 battery dataset. The dataset utilized in our analysis was obtained from the Hawaii Natural Energy Institute, which conducted the experiments and collected the data. The experimental data included measurements from 14 NMC-LCO 18650 batteries with a nominal capacity of 2.8 Ah. The batteries were subjected to cycling tests over 1000 cycles at a temperature of 25°C. During the tests, the batteries were charged using a CC-CV (Constant Current-Constant Voltage) charge rate of C/2 and discharged at a rate of 1.5C. To ensure our predictive model's reliability and generalization capability, we employ a separate testing set consisting of batteries that are not part of the training process. The decision to use separate training and testing sets holds practical significance for predicting the RUL of batteries. By training the model on a subset of batteries and evaluating its performance on unseen batteries, we aim to assess its ability to generalize and make accurate predictions for batteries that have not been encountered during the training phase.

A. DATA DESCRIPTION

The data was collected under various conditions, such as different loading profiles of 1A, 2A, and 4A, shifting ambient temperatures of 4°C, room temperature, and 44°C, and discharge voltages of 2.7V, 2.5V, and 2.2V. The external temperatures, ranging from 4°C to 44°C, were also monitored in addition to the battery temperatures. The total capacity fluctuates between 2Ah and 0Ah in each charging and discharging cycle. It is important to note that deep discharge can harm the battery, and as a result, the battery voltage is typically between 3.7V and 4.2V during operation.

B. DATA PROCESSING

In addition to the initial data cleaning steps, several data processing techniques were employed to prepare the dataset for analysis. These included feature engineering, normalization, and data partitioning. In order to create new features from the raw data and extract pertinent information, feature engineering was used. In order to ensure that all features were on an equal scale, normalization was used, which can enhance the performance of ML algorithms. The features were scaled precisely to have a unit variance and a zero mean. The training, validation, and test sets were then created from the dataset. Model selection and hyperparameter tweaking were made on the validation set, final model performance was evaluated on the test set, and ML models were fitted on the training set. In order to make sure that each set had a balanced representation of the battery cycles, the data was stratified and divided.

In order to produce predictions, the dataset's several fields must be analyzed. The operations of feature extraction, selection, normalization, and segmentation are included in data preparation. A phase in data preparation is the creation of

TABLE 2. Distribution of samples between training and testing sets.

Data	Rows	Percentage(%)
Total data	2500	100
Training data	1625	65
Test data	875	35

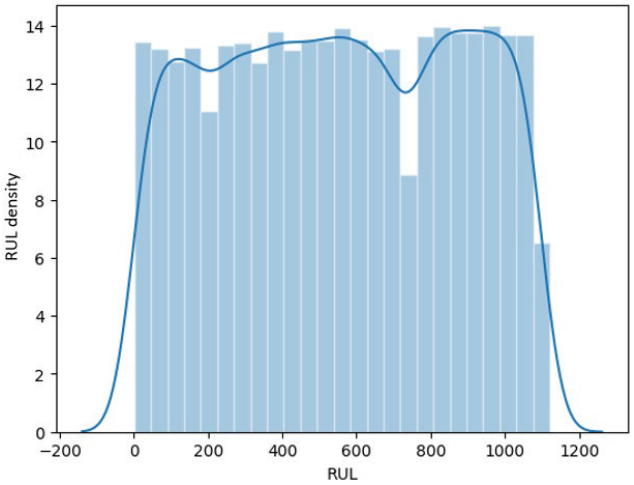

FIGURE 5. RUL Histogram of Lithium-Ion Batteries.

training, testing, and validation datasets. Using the battery dataset, we divided it into three categories: training, validation, and testing, using 65% training data and 35% testing data. Table 2 displays the dataset partition that considers training and testing.

V. RESULTS AND DISCUSSION

This section evaluates the proposed integrated process's outcomes and implementation strategy, including performance evaluation and ML-based RUL prediction. RUL is the estimated remaining lifespan of a machine or component before it becomes inoperable or fails to meet performance criteria. RUL prediction is essential to PdM to ensure machines' reliable and safe operation and reduce maintenance costs. Figure 5 shows the Remaining Useful Time (RUT) histogram representation of the statistical distribution of the data considering only the battery life, in which the y-axis represents RUL density or frequency. The x-axis represents RUL. The frequency of each value refers to the number of occurrences or the proportion of data points with a specific value. In the context of the battery life represented on the graph, the frequency of each value indicates how often a particular battery life duration occurs in the dataset. Higher frequencies suggest that certain battery life durations are more common, while lower frequencies indicate less common durations. In fact, the curves represent the distribution, and the frequency of each value on each curve indicates the frequency of each value. It is a valuable tool for visualizing the distribution of RUL values and identifying trends or patterns in the data. The RUT histogram can provide insights into the health status of machines and help maintenance teams plan maintenance activities and predict failures.

Table 3 outlines the technology and tools used in the implementation and the necessary settings. The operative technique on which the strategy was implemented and run as Windows 10 utilizing an Intel(R) CPU Core(TM), i5-9600K, at 3.70 GHz, the RAM used was 16 GB, and the programming language was Python 3.8.3.

TABLE 3. Proposed approach tools and strategies of implementation.

No	System Component	Description
1	Operating System	Windows 10
2	CPU	Intel(R) Core(TM) i5-9600K
3	Programming Language	Python 3.8.3
4	CPU GHz	3.70
5	Browser	Google Chrome
6	RAM	16 G

A. FEATURE SELECTION

The most important characteristics that can aid in predicting the batteries' RUL were identified during the feature selection process. Selecting relevant features (variables, predictors) from a more extensive set of features is known as feature selection. By reducing overfitting, shortening training time, and enhancing the model's interpretability, feature selection seeks to improve the model's performance. Filter, wrapper, and embedding methods are the three main categories into which feature selection techniques can be divided. It was accomplished using various feature selection techniques, such as feature significance analysis using RF, mutual information analysis, and correlation analysis.

B. EVALUATION OF RESULTS WITH DIFFERENT METRICS

In order to assess the effectiveness and accuracy of our method in predicting the RUL of the batteries, we used several metrics. The metrics we used include Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Squared Error (MSE), coefficient of determination (R^2), Mean Absolute Percentage Error (MAPE), Normalized Mean Squared Error (NMRSE), and correlation coefficient. The MAE, MSE, and RMSE measure the absolute, squared, and mean squared differences between the actual and predicted RUL values. MAPE also shows the percentage difference between the actual and anticipated RUL values. The correlation coefficient reflects the linear connection between the actual and anticipated RUL values, whereas NMRSE is a normalized measure of the RMSE that considers the data's unpredictability. These metrics provide a comprehensive evaluation of the performance of our method and enable us to compare its effectiveness with other techniques in the literature.

1) THE MEAN ABSOLUTE ERROR

MAE is a generally utilized metric to evaluate the performance of regression models. In predicting the RUL of batteries, MAE can be used to measure the average absolute difference between the predicted RUL and the valid RUL values. The MAE formula is defined as in Equation 5:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |RUL_{i,actual} - RUL_{i,predicted}| \qquad (5)$$

where n describes the total number of samples, $RUL_{i,actual}$ represents the actual RUL value for the i^{th} sample, and $RUL_{i,predicted}$ defines the predicted RUL value for the i^{th} sample. The MAE is the average absolute difference between the true and predicted RUL values across all batteries in the dataset.

2) ROOT MEAN SQUARED ERROR

A common metric for assessing the precision of a predictive model is the RMSE. It computes the average squared difference between a given variable's predicted and actual values. In the case of predicting RUL, the RMSE formula can be written as in Equation 6:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(RUL_{i,actual} - RUL_{i,predicted})^2} \quad (6)$$

where n defines the number of samples, $RUL_{i,actual}$ is the actual RUL for sample i, and $RUL_{i,predicted}$ describes the predicted RUL for sample i. The RMSE penalizes more significant errors more heavily than minor ones, as it squares the differences before taking the square root. A lower RMSE indicates better predictive performance, as the predicted values are closer to the actual values.

3) MEAN SQUARED ERROR

The RUL can be defined in Equation 7, and the MSE calculates the average of the squared differences between the predicted and actual RUL values.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(RUL_{i,actual} - RUL_{i,predicted})^2 \quad (7)$$

The MSE is a statistic frequently used to assess the effectiveness of regression models since it calculates the average squared difference between the actual and projected RUL values.

4) MEAN ABSOLUTE PERCENTAGE ERROR

The MAPE is a common metric for evaluating prediction accuracy, particularly when utilizing time series data. It determines the average difference between the predicted and actual values as a percentage of the actual values. The formula for MAPE in terms can be expressed as in Equation 8:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{RUL_{i,actual} - RUL_{i,predicted}}{RUL_{i,actual}}\right| \times 100\% \quad (8)$$

where n defines the number of observations, $RUL_{i,actual}$ describes the RUL for the ith comment, and $RUL_{i,predicted}$ describes the predicted RUL for the ith observation. The MAPE is expressed as a percentage and helps compare the accuracy of different models or methods for predicting RUL. However, it is essential to note that the MAPE has some limitations, such as its sensitivity to extreme values and the fact that it can be undefined if any actual values are zero. Therefore, it is often used with other metrics to provide a complete evaluation of the performance of RUL prediction models.

5) NORMALIZED MEAN SQUARED ERROR

The sum of the squared deviations between the actual and expected RUL values is shown in the numerator. The denominator represents the total squared errors between the real RUL values and their mean. The NRMSE measures the accuracy of the predicted RUL values, normalized by the variability of the actual RUL values. A lower NRMSE indicates better accuracy and predictive power. The NRMSE formula in terms of RUL is performed as follows in Equation 9:

$$NRMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{RUL_{i,actual} - RUL_{i,predicted}}{RUL_{i,max} - RUL_{i,min}}\right)^2} \quad (9)$$

where n represents the number of observations, $RUL_{i,actual}$ represent the actual RUL value for observation i, $RUL_{i,predicted}$ defines the predicted RUL value for observation i, $RUL_{i,max}$ represent the maximum RUL value in the dataset, and $RUL_{i,min}$ defines the minimum RUL value in the dataset. The NRMSE value ranges from 0 to 1, with a smaller value indicating better performance. A value of 0 indicates a perfect match between the predicted and actual values, while a value of 1 indicates that the predicted values are entirely uncorrelated with the actual values.

6) R^2

A statistical measure known as the coefficient of determination, or R^2, quantifies the percentage of the dependent variable's variance that can be predicted from the independent variable (s). It is frequently used to gauge the goodness of fit of a regression model. In the context of predicting RUL of batteries, R^2 is calculated as indicated in Equation Equation 10:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(RUL_{i,actual} - RUL_{i,predicted})^2}{\sum_{i=1}^{n}(RUL_{i,actual} - \bar{RUL}_{actual})^2} \quad (10)$$

where n is the number of samples, $RUL_{i,actual}$ defines the actual RUL value for the ith sample, $RUL_{i,predicted}$ represents the predicted RUL value for the ith sample, and $\bar{RUL}_{actual}$ is the mean actual RUL value across all samples. The range of $R2$ is 0 to 1, where a value of 0 implies that the model does not explain any variability in the dependent variable, and a value of 1 suggests that it does. The real and anticipated RUL values fit more closely when the $R2$ value is larger [33].

The Table 4 summarizes the performance of two ML models, RFt, and LightGBM, in predicting the RUL of batteries. The models were optimized using HHO for hyperparameter tuning. The models' effectiveness was assessed using a variety of metrics, including MAE, MSE, RMSE, MAPE, R^2, and NMRSE. The table's first two rows show the RF model's training and testing scores. The model achieved a low MAE of 36.883 and a low MSE of 2148.865 for the training data, indicating that the model's predictions were close to

TABLE 4. Performance evaluation of the proposed method for predicting the RUL of batteries.

Random Forest Optimization With HHO	MAE	MSE	RMSE	MAPE	R2	NMRSE
Training Scores	36.883	2148.865	46.356	0.239	0.979	0.146
Testing Scores	49.306	3680.922	60.671	0.483	0.969	0.206
LightGBM Optimization With HHO	MAE	MSE	RMSE	MAPE	R2	NMRSE
Training Scores	37.681	2209.121	47.001	0.255	0.979	0.148
Testing Scores	47.951	3512.709	59.268	0.473	0.971	0.202

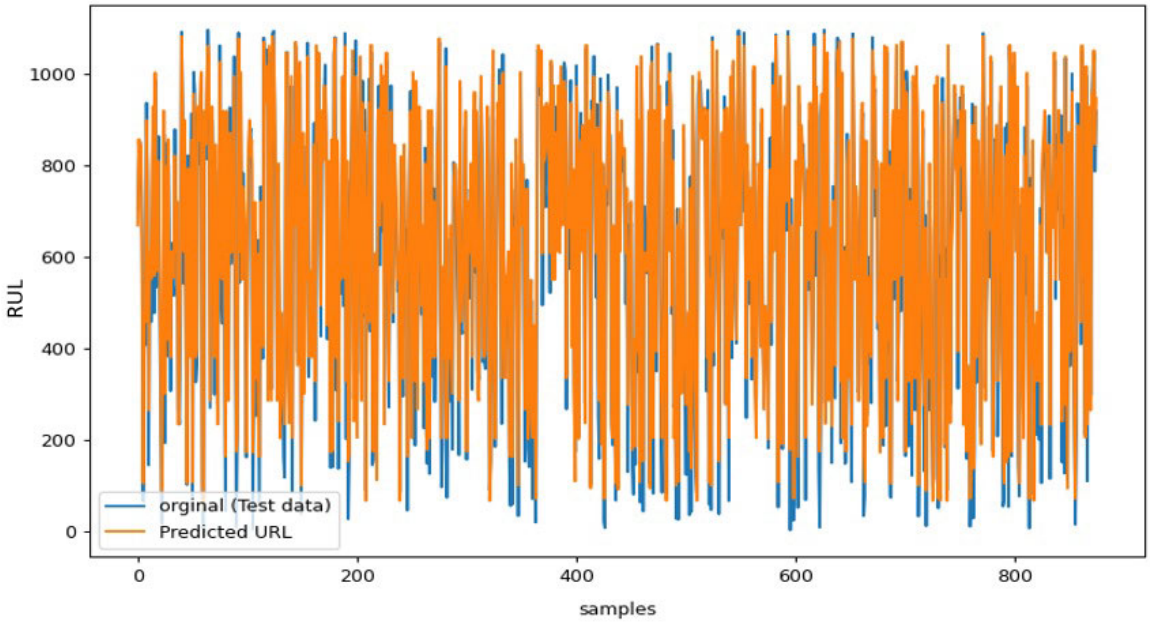

FIGURE 6. RUL battery dataset with predicted and actual values.

the actual RUL values. The RMSE value of 46.356 indicates that the average prediction error was small. The model also achieved a low MAPE of 0.239, which means that the model's average prediction error was only 0.239 of the actual RUL value. The R^2 score of 0.979 indicates that the model's predictions fit the training data well. The NMRSE score of 0.146 indicates a good fit of the predictions to the actual data. The third and fourth rows of the table 4 show the training and testing scores for the LightGBM model. The model achieved a low MAE of 37.681 and a low MSE of 2209.121 for the training data, indicating that the model's predictions were close to the actual RUL values. The RMSE value of 47.001 indicates that the average prediction error was small. The model also achieved a low MAPE of 0.255, which means that the model's average prediction error was only 0.255 of the actual RUL value. The R^2 score of 0.979 indicates that the model's predictions fit the training data well. The NMRSE score of 0.148 indicates a good fit of the predictions to the actual data. In general, both models achieved high accuracy in predicting RUL, as evidenced by the low values of the evaluation metrics.

Figure 6 provides the actual and predicted values by the model. Our goal is to demonstrate the model's performance and how accurately it can make predictions. The y-axis represents RUL, and the x-axis represents the sample because we have selected a set of samples to test the model. The blue data points represent the actual values, and the orange data points represent our predictions, which closely match the data in many points. This indicates a good performance of the model. The values may deviate more or less in some points, and the model's error can be calculated based on these differences.

TABLE 5. Comparison of MAPE and RMSE values for various RUL prediction methods.

Methods	MAPE	RMSE
CNN-LSTM [37]	6.43	-
PSR-SVR [38]	20.3	191.83
GRU-RNN [39]	9.94	127.65
CNN-LSTM [40]	14.85	176.63
Proposed Method	0.23	46.35

Figure 7 displays the predicted results for the RUL battery prediction, showcasing a comparison between our model's actual and predicted values. The blue line shows the true RUL values, while the orange line shows the corresponding predicted RUL values. By analyzing Figure7 in conjunction with Figure 6, we gain deeper insights into the performance of our RUL prediction model. The comparison between the actual and predicted values allows us to assess the accuracy and reliability of our method. Furthermore, the close alignment of the blue and orange lines in Figure 7 indicates high accuracy in our RUL predictions. The model's ability to closely track the actual RUL values demonstrates its effectiveness in forecasting battery RUL.

TABLE 6. The list of abbreviations and formula symbols.

Nomenclature	
Symbols	
I	Current
t	Time
$x_{i,j}(t)$	The j-th component of the position of the i-th hawk at time t
$x_{best,j}(t)$	The best position found so far in the search space
r_1, r_2	Random coefficients between 0 and 1
$x_{p,j}(t), x_{q,j}(t)$	Positions of two randomly selected hawks
$\hat{y}$	Predicted response value
Θ	Set of decision trees
M	Number of decision trees in the ensemble
N	Number of samples in the dataset
X_{ij}	Value of the j-th feature for the i-th sample
y_i	Label corresponding to the i-th sample
l	Loss function
f_k	k-th decision tree
K	Total number of trees
$\Omega(f_k)$	Regularization term penalizing model complexity
$Obj(\Theta)$	Objective function to be minimized
λ	Regularization parameter
γ	Regularization parameter
Δ	Split gain
G_L	Gradient statistics for the potential split's left node
G_R	Gradient statistics for the potential split's right node
H_L	The Hessian matrix (for the left node)
H_R	The Hessian matrix (for the right node)
$RUL_{i,actual}$	The actual Remaining Useful Life (RUL) value for the i-th data point.
$RUL_{i,predicted}$	The predicted RUL value for the i-th data point.
Acronyms	
EV	Electric Vehicle
Ah	Ampere hour
SOH	State Of Health
BMS	Battery Management System
ML	Machine learning
RUL	Remaining Useful Life
NMC-LCO	Nickel Manganese Cobalt Lithium Cobalt Oxide
PdM	Predictive Maintenance
MAE	Mean Absolute Error
MSE	Mean Squared Error
MAPE	Mean Absolute Percentage Error
RMSE	Root Mean Squared Error
R^2	Coefficient of Determination
NMRSE	Normalized Mean Squared Error
RF	Random Forest
LightGBM	Light Gradient Boosting Machine
HHO	Harris Hawks Optimization
Obj	Objective function
SGD	Stochastic Gradient Descent
Hessian	In mathematics and optimization, the Hessian refers to the matrix of second-order partial derivatives of a scalar-valued function
CC-CV	Constant Current-Constant Voltage
C/2	The charging or discharging rate relative to the battery's nominal capacity

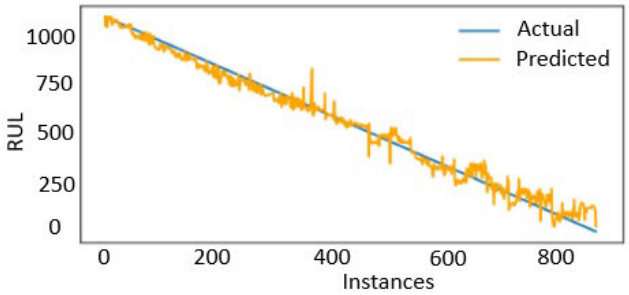

FIGURE 7. Prediction of RUL for the battery.

The x-axis in the R-Squared (Testing Data) Figure represents the predicted RUL values, while the y-axis represents the measured RUL values. The blue dots in the figure represent the predicted RUL values for each sample. In contrast, the diagonal line represents the perfect prediction line where the predicted RUL values match the measured RUL values. The R^2 score of 0.971 indicates that the proposed model based on RF and Lightgbm optimization with HHO achieved a high level of accuracy in predicting the RUL of batteries, with the predicted values closely matching the measured values. These outcomes illustrate the efficiency of the suggested approach and its potential to enhance battery RUL prediction. The proposed method offers increased accuracy in RUL battery prediction, providing an effective solution for predictive maintenance in the field of battery technology.

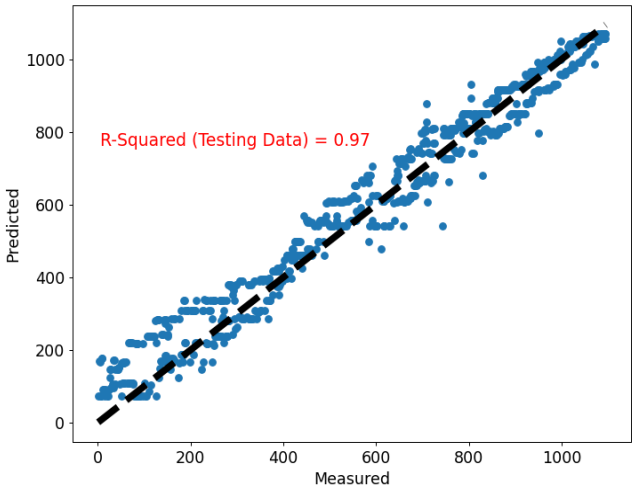

FIGURE 8. The R-Squared of RUL predictions using proposed model based on RF and LightGBM optimization with HHO.

Table 5 compares the R^2 values achieved by four different methods for predicting the RUL of batteries. The first row shows the reference method, the Unscented Kalman Filter (UKF), used by Sangwan et al. [34]. The second row shows the hybrid method Cong et al. [35] used, which combines machine learning and statistical models. The third row shows the Empirical-Data Hybrid-Driven Approach (EDHDA) proposed by Chen et al. [36]. The fourth and final row shows the R^2 value achieved by the proposed method in this study, which uses RF and LightGBM optimization with HHO for hyperparameter tuning. The R^2 values range from 0.84 to 0.97%, with the proposed method achieving the highest R^2 value of 0.97%, indicating a high level of accuracy in predicting the RUL of batteries.

Table 5 presents a comparison of MAPE and RMSE values for different methods used in predicting the RUL of batteries. Our proposed method outperforms all other methods with significantly lower MAPE (0.23) and RMSE (46.35) values. These results highlight our approach's superior accuracy and precision, showcasing its potential for accurate RUL prediction and ensuring lithium-ion batteries' reliable and efficient operation in various applications.

VI. CONCLUSION AND FUTURE WORK

RUL prediction plays a crucial role in ensuring systems' overall reliability and safety, which is essential for developing successful maintenance strategies. ML techniques have been increasingly used in various fields to predict the RUL of assets, and historical data can be leveraged to predict RUL, enabling efficient maintenance planning and cost savings associated with preventing equipment failures. The proposed approach in this study, which combines RF and LightGBM models with HHO and Sensor Fusion, effectively predicts the RUL of Lithium-Ion Batteries, outperforming traditional approaches. The HHO algorithm provided a better search space for the models' hyperparameters, resulting in improved prediction accuracy. The incorporation of sensor fusion helped extract more meaningful features from the battery data, further enhancing RUL prediction accuracy. The high R^2 values for both training and testing data suggest that the proposed approach is highly effective in predicting the RUL of batteries. Thus, the proposed approach can be a valuable tool for PdM of Lithium-Ion Batteries in various applications, including EVs and renewable energy systems. However, there is still room for improvement, and future work could focus on exploring other optimization algorithms and feature selection techniques to enhance the RUL prediction's accuracy further. Additionally, expanding the scope of the study to include other types of batteries and different operating conditions could provide a more comprehensive understanding of the proposed approach's effectiveness.

Table 6 displays the primary notations utilized in the suggested strategy.

REFERENCES

[1] C. Depcik, T. Cassady, B. Collicott, S. P. Burugupally, X. Li, S. S. Alam, J. R. Arandia, and J. Hobeck, "Comparison of lithium ion batteries, hydrogen fueled combustion engines, and a hydrogen fuel cell in powering a small unmanned aerial vehicle," *Energy Convers. Manag.*, vol. 207, Mar. 2020, Art. no. 112514.

[2] Z. Liu, B. He, Z. Zhang, W. Deng, D. Dong, S. Xia, X. Zhou, and Z. Liu, "Lithium/graphene composite anode with 3D structural LiF protection layer for high-performance lithium metal batteries," *ACS Appl. Mater. Interfaces*, vol. 14, no. 2, pp. 2871–2880, Jan. 2022.

[3] A. J. C. Trappey, C. V. Trappey, U. H. Govindarajan, A. C. Chuang, and J. J. Sun, "A review of essential standards and patent landscapes for the Internet of Things: A key enabler for Industry 4.0," *Adv. Eng. Informat.*, vol. 33, pp. 208–229, Aug. 2017.

[4] L. Zheng, Y. He, L. Fan, H. Cao, and X. Chen, "Semi-active vibration control of the motorized spindle using a self-powered SSDV technique: Simulation and experimental study," *Automatika*, vol. 63, no. 3, pp. 511–524, Jul. 2022.

[5] M. Baptista, E. M. P. Henriques, I. P. de Medeiros, J. P. Malere, C. L. Nascimento, and H. Prendinger, "Remaining useful life estimation in aeronautics: Combining data-driven and Kalman filtering," *Rel. Eng. Syst. Saf.*, vol. 184, pp. 228–239, Apr. 2019.

[6] E. Uhlmann, R. P. Pontes, C. Geisert, and E. Hohwieler, "Cluster identification of sensor data for predictive maintenance in a selective laser melting machine tool," *Proc. Manuf.*, vol. 24, pp. 60–65, Jan. 2018.

[7] A. L. Ellefsen, E. Bjørlykhaug, V. Æsøy, S. Ushakov, and H. Zhang, "Remaining useful life predictions for turbofan engine degradation using semi-supervised deep architecture," *Rel. Eng. Syst. Saf.*, vol. 183, pp. 240–251, Mar. 2019.

[8] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long short-term memory network for remaining useful life estimation," in *Proc. IEEE Int. Conf. Prognostics Health Manag. (ICPHM)*, Jun. 2017, pp. 88–95.

[9] A. Kanawaday and A. Sane, "Machine learning for predictive maintenance of industrial machines using IoT sensor data," in *Proc. 8th IEEE Int. Conf. Softw. Eng. Service Sci. (ICSESS)*, Nov. 2017, pp. 87–90.

[10] X. Li, Q. Ding, and J.-Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Rel. Eng. Syst. Saf.*, vol. 172, pp. 1–11, Apr. 2018.

[11] Y. Che, X. Hu, X. Lin, J. Guo, and R. Teodorescu, "Health prognostics for lithium-ion batteries: Mechanisms, methods, and prospects," *Energy Environ. Sci.*, vol. 16, no. 2, pp. 338–371, 2023.

[12] C. Zhang, S. Zhao, and Y. He, "An integrated method of the future capacity and RUL prediction for lithium-ion battery pack," *IEEE Trans. Veh. Technol.*, vol. 71, no. 3, pp. 2601–2613, Mar. 2022.

[13] S. Zhao, C. Zhang, and Y. Wang, "Lithium-ion battery capacity and remaining useful life prediction using board learning system and long short-term memory neural network," *J. Energy Storage*, vol. 52, Aug. 2022, Art. no. 104901.

[14] C. Zhang, S. Zhao, Z. Yang, and Y. Chen, "A reliable data-driven state-of-health estimation model for lithium-ion batteries in electric vehicles," *Frontiers Energy Res.*, vol. 10, Sep. 2022, Art. no. 1013800.

[15] Y. Che, D.-I. Stroe, X. Hu, and R. Teodorescu, "Semi-supervised self-learning-based lifetime prediction for batteries," *IEEE Trans. Ind. Informat.*, vol. 19, no. 5, pp. 6471–6481, May 2023.
[16] Y. Che, Z. Deng, X. Tang, X. Lin, X. Nie, and X. Hu, "Lifetime and aging degradation prognostics for lithium-ion battery packs based on a cell to pack method," *Chin. J. Mech. Eng.*, vol. 35, no. 1, pp. 1–16, Dec. 2022.
[17] Y. Che, Z. Deng, X. Lin, L. Hu, and X. Hu, "Predictive battery health management with transfer learning and online model correction," *IEEE Trans. Veh. Technol.*, vol. 70, no. 2, pp. 1269–1277, Feb. 2021.
[18] R. Wang, M. Zhu, X. Zhang, and H. Pham, "Lithium-ion battery remaining useful life prediction using a two-phase degradation model with a dynamic change point," *J. Energy Storage*, vol. 59, Mar. 2023, Art. no. 106457.
[19] Y. Liu, J. Sun, Y. Shang, X. Zhang, S. Ren, and D. Wang, "A novel remaining useful life prediction method for lithium-ion battery based on long short-term memory network optimized by improved sparrow search algorithm," *J. Energy Storage*, vol. 61, May 2023, Art. no. 106645.
[20] G. Lou, W. Lin, G. Huang, and W. Xiang, "A two-stage online remaining useful life prediction framework for supercapacitors based on the fusion of deep learning network and state estimation algorithm," *Eng. Appl. Artif. Intell.*, vol. 123, Aug. 2023, Art. no. 106399.
[21] L. Li, Z. Wang, and H. Jiang, "Storage battery remaining useful life prognosis using improved unscented particle filter," *Proc. Inst. Mech. Eng., O, J. Risk Rel.*, vol. 229, no. 1, pp. 52–61, Feb. 2015.
[22] L. Yan, J. Peng, D. Gao, Y. Wu, Y. Liu, H. Li, W. Liu, and Z. Huang, "A hybrid method with cascaded structure for early-stage remaining useful life prediction of lithium-ion battery," *Energy*, vol. 243, Mar. 2022, Art. no. 123038.
[23] X. Li, L. Zhang, Z. Wang, and P. Dong, "Remaining useful life prediction for lithium-ion batteries based on a hybrid model combining the long short-term memory and Elman neural networks," *J. Energy Storage*, vol. 21, pp. 510–518, Feb. 2019.
[24] A. F. Hofmann, D. N. Fronczek, and W. G. Bessler, "Mechanistic modeling of polysulfide shuttle and capacity loss in lithium–sulfur batteries," *J. Power Sources*, vol. 259, pp. 300–310, Aug. 2014.
[25] M. Ahwiadi and W. Wang, "An enhanced mutated particle filter technique for system state estimation and battery life prediction," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 3, pp. 923–935, Mar. 2019.
[26] Y. Li, K. Liu, A. M. Foley, A. Zülke, M. Berecibar, E. Nanini-Maury, J. Van Mierlo, and H. E. Hoster, "Data-driven health estimation and lifetime prediction of lithium-ion batteries: A review," *Renew. Sustain. Energy Rev.*, vol. 113, Oct. 2019, Art. no. 109254.
[27] L. Zhang, X. Hu, Z. Wang, F. Sun, J. Deng, and David. G. Dorrell, "Multiobjective optimal sizing of hybrid energy storage system for electric vehicles," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1027–1035, Feb. 2018.
[28] A. Guha and A. Patra, "Online estimation of the electrochemical impedance spectrum and remaining useful life of lithium-ion batteries," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 8, pp. 1836–1849, Aug. 2018.
[29] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019.
[30] H. M. Alabool, D. Alarabiat, L. Abualigah, and A. A. Heidari, "Harris hawks optimization: A comprehensive review of recent variants and applications," *Neural Comput. Appl.*, vol. 33, no. 15, pp. 8939–8980, Aug. 2021.
[31] L. Yin, Z. Sun, F. Gao, and H. Liu, "Deep forest regression for short-term load forecasting of power systems," *IEEE Access*, vol. 8, pp. 49090–49099, 2020.
[32] H. Liu, Q. Xiao, Y. Jin, Y. Mu, J. Meng, T. Zhang, H. Jia, and R. Teodorescu, "Improved LightGBM-based framework for electric vehicle lithium-ion battery remaining useful life prediction using multi health indicators," *Symmetry*, vol. 14, no. 8, p. 1584, Aug. 2022.
[33] H. Kaneko, "Beware of r^2 even for test datasets: Using the latest measured y-values (r^2_{LM}) in time series data analysis," *J. Chemometrics*, vol. 33, no. 2, p. e3093, Feb. 2019.
[34] V. Sangwan, R. Kumar, and A. K. Rathore, "An empirical capacity degradation modeling and prognostics of remaining useful life of Li-ion battery using unscented Kalman filter," in *Proc. 8th IEEE India Int. Conf. Power Electron. (IICPE)*, Dec. 2018, pp. 1–6.
[35] X. Cong, C. Zhang, J. Jiang, W. Zhang, and Y. Jiang, "A hybrid method for the prediction of the remaining useful life of lithium-ion batteries with accelerated capacity degradation," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 12775–12785, Nov. 2020.
[36] D. Chen, J. Meng, H. Huang, J. Wu, P. Liu, J. Lu, and T. Liu, "An empirical-data hybrid driven approach for remaining useful life prediction of lithium-ion batteries considering capacity diving," *Energy*, vol. 245, Apr. 2022, Art. no. 123222.
[37] Y. Zhang, R. Xiong, H. He, and M. G. Pecht, "Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 5695–5705, Jul. 2018.
[38] L. Chen, Y. Zhang, Y. Zheng, X. Li, and X. Zheng, "Remaining useful life prediction of lithium-ion battery with optimal input sequence selection and error compensation," *Neurocomputing*, vol. 414, pp. 245–254, Nov. 2020.
[39] R. R. Ardeshiri and C. Ma, "Multivariate gated recurrent unit for battery remaining useful life prediction: A deep learning approach," *Int. J. Energy Res.*, vol. 45, no. 11, pp. 16633–16648, Sep. 2021.
[40] G. Ma, Y. Zhang, C. Cheng, B. Zhou, P. Hu, and Y. Yuan, "Remaining useful life prediction of lithium-ion batteries based on false nearest neighbors and a hybrid neural network," *Appl. Energy*, vol. 253, Nov. 2019, Art. no. 113626.

SADIQA JAFARI received the B.S. degree in electric engineering from Pooyesh University, Iran, and the M.S. degree in control engineering from the University of Qom, Qom, Iran, in 2019. In February 2021, she moved to Jeju-do, Republic of Korea, for her Ph.D. studies and started working as a Ph.D. Research Fellow with the Machine Learning Laboratory (MLL), Jeju National University. Her research interests include energy systems, model predictive control, fuzzy control, multi-agent systems, artificial intelligence, machine learning, natural language processing, and data mining.

YUNG-CHEOL BYUN received the B.S. degree from Jeju National University, in 1993, and the M.S. and Ph.D. degrees from Yonsei University, in 1995 and 2001, respectively. He was a Special Lecturer with SAMSUNG Electronics and SDS, from 1998 to 2001. From 2001 to 2003, he was a Senior Researcher with the Electronics and Telecommunications Research Institute (ETRI). He was promoted to join Jeju National University as an Assistant Professor, in 2003, where he is currently a Full Professor with the Computer Engineering Department. His research interests include AI and machine learning, pattern recognition, blockchain and deep learning-based applications, big data and knowledge discovery, time-series data analysis and prediction, image processing, medical image applications, and recommendation systems.