

Original Article

GT-SOH: A Graph-Transformer Contrastive Learning Framework with Starfish Optimization for Accurate State-of-Health Battery Prediction in EV

R. Natarajan¹, Jeya Prakash Kadambarajan², K. Selva Sheela³, S. Karthik⁴

¹Department of Electrical and Electronics Engineering, Fatima Michael College of Engineering and Technology, Madurai, Tamil Nadu, India

²Department of Electronics and Communication Engineering, Kalasalingam Academy of Research and Education (Kalasalingam University), Krishnankoil, Tamil Nadu, India.

³Department of Computer Science and Engineering, VELTECH Rangarajan Dr.Sakunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India.

⁴Department of Electronics and Communication Engineering, SRM Madurai College for Engineering and Technology, Sivagangai, Tamil Nadu, India.

¹Corresponding Author : natarajanfmcteee@gmail.com

Received: 14 August 2025

Revised: 16 September 2025

Accepted: 15 October 2025

Published: 31 October 2025

Abstract - In specific, accurate, and precise prediction of the State-of-Health (SOH) of the Electric Vehicle (EV) battery system will be crucial to ensure battery safety, longevity, and optimal management of energy. Existing SoH estimation models, which include Machine Learning (ML) and Deep Learning (DL) schemes, most often struggle in capturing the most complex temporal and spatial dependencies in the degradation of the battery. In this paper, a novel Graph-Transformer Contrastive Learning (GT-SoH) framework, which incorporates Graph Neural Networks (GNNs) termed Transformer-based temporal modeling, Contrastive self-learning, and Starfish Optimization Algorithm (SFOA) for hyperparameter tuning, is proposed and is denoted as the (GT-SOH-SFOA) framework. A GNN model is responsible for capturing spatial interdependencies among battery cells, whereas a Transformer encoder models GNN patterns. A contrastive learning function is used for enhancing the generalizability of learning a robust representation of features from unlabeled battery datasets. In addition, SFOA is employed to tune the hyperparameters, thus ensuring optimal performance for balancing exploitation and exploration in the process of optimization. The hybrid loss function, which integrates Mean Absolute Error (MAE) loss and contrastive loss, ensures precise SOH estimation, thus reducing overfitting. An experimental evaluation is carried out for various metrics like Mean Absolute Error (MAE), R^2 , RMSE, and Max Error on four datasets, like Musoshi, NASA, Stanford, and the BMW i3 battery dataset, and outcomes attained demonstrate that the GT-SOH-SFOA proposed model outperforms existing models compared, thereby offering high prediction accuracy and robustness. Therefore, it is concluded that the proposed scheme offers a scalable, interpretable, and optimized solution for real – time battery health monitoring in EVs.

Keywords - State-of-Health (SOH), Electric Vehicle (EV), Battery health prediction, Graph-Transformer Contrastive Learning, Starfish optimization algorithm, Graph Neural Networks.

1. Introduction

Batteries are usually critical for a varied range of applications, which include portable electronic devices and EVs. The optimization and monitoring of battery performance are essential for the efficiency and safety of those applications [1]. Battery Management System (BMS) has become a major point of academic research and industry [2-4]. Predictions of battery health and SoC are highly needed in real-time applications, which impact the optimization range of EV, extension of battery lifespan, and management of energy. Moreover, the traditional BMS systems have some limitations

because of the complex chemical processes and the aging of the battery. A dynamic chemical variation and aging of the battery component could introduce some errors in the process of charge prediction. In addition, the absence of perfect sensors makes precise measurement of all variables, which in turn reveals the measurement methods directly [5-7]. Those limitations result from the measurement issue caused by external factors like the noise sensor, which could cause a misleading outcome. To deal with these limitations, researchers are exploring several kinds of models for increasing the health of the battery and the accuracy of the



prediction of SoC. SoH is regarded as a metric that enumerates the level of battery degradation in relation to a new battery. This information is needed for the management of energy systems in vehicles so as to fine-tune their control and thereby maintain the performance of vehicles and safety within the constrained limits. Several models could be employed to assess and measure the SOH of batteries in EVs. Much research at present focuses primarily on identifying the decrease in capacity (SOHr) [8]. Typically, deterioration of Lithium-Ion Batteries (LIBs) will be unavoidable during storage or cycling. The SOH data of the battery is needed for EVs' energy management system so as to confirm exceptionally efficient & secured functional state [9-11]. The examination and analysis mechanism causes batteries to deteriorate over time, thus resulting in effects that are vital for measuring the health of batteries accurately and making dependable performance forecasting. The primary intention is thus to make the battery system a highly reliable and effective one while extending the life span of the battery. Thus, data-driven modeling solutions become valuable, particularly Artificial Intelligence (AI) and Machine Learning (ML) models, which could learn from data and thereby generalize the unobservable factors.

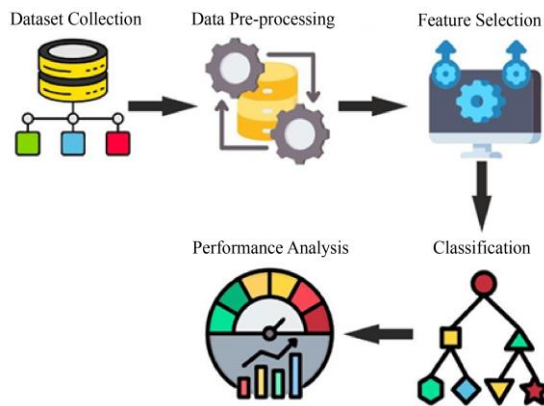


Fig. 1 Typical flow of SOH prediction

In contrast to the existing schemes, ML schemes recognize high-dimensional data patterns, independent of fundamental physical processes. It, in turn, makes the study unaffected by factors like hard-to-study, thereby making ML less favorable for the estimation of battery state [12]. A Graph-transformer contrastive learning model's ability for modeling the time-series dependencies effectually than existing models, which aims at achieving highly accurate predictions of SOH. A self-attention model designed to adopt the instantaneous alteration enables the transformer to handle real-time and sudden alterations in the dynamics of the battery. This feature could enhance the accuracy of SOC prediction under charging/discharging conditions. The transformer model's ability to capture non-linear patterns and temporal dependencies in the cycles of battery charge/discharge conditions. A transformer model thus offers an innovative solution for the challenges where the traditional models fall

short, like sensor noise, aging effect, and dynamic chemical processes. Their superior accuracy in the prediction of SOH signifies the significant innovation aiming at optimizing the range of EV, battery life, and charging efficiency. The utilization of a dependent model in the estimation of SOH, along with an optimization strategy, potentially sets a new standard in monitoring the health of the battery.

1.1. Problem Identification / Motivation

This model addresses the existing issue of SOH prediction with the growth in EV adoption. Though several ML and DL models exist, there are some issues like low prediction and poor performance, and there is a need to enhance feature weighting and the accurate and robust creation of SOH estimation [13]. Addressing these gaps might contribute to the development of SOH estimation models, thus ultimately supporting effective and reliable BMSs for a varied range of applications, including EV. By incorporating real-time data, this model offers a substantial contribution to comparing research limited to smaller or simultaneous datasets. This work highlights the benefits of several scenarios.

1.2. Contribution

The major contributions of this work on SOH prediction using an optimization-based deep learning model are listed as follows:

- To implement and design a new Graph-Transformer Contrastive Learning (GT-SoH) framework which integrates Graph Neural Networks (GNNs), Transformer-based temporal modeling, contrastive self-supervised learning, and Starfish Optimization Algorithm (SFOA) for improving the accuracy of SOH prediction.
- To preprocess input data using min-max normalization to ensure that all features will be scaled between 0 and 1, thus enabling the training model to work effectively.
- For estimating the capability of the proposed Graph-Transformer Contrastive Learning based temporal model for capturing temporal patterns and dependencies in the performance of battery data, thus enhancing the prediction accuracy of SOH.
- To improve the process of feature selection with the use of the SFOA optimization model by selecting more relevant features to be used in the prediction model of SOH.
- To assess the suggested GT-SoH-SFOA scheme with the use of key performance metrics and to validate the suggested model efficiency over other existing models by making a comparison between the outcomes.

1.3. Organization

This research work includes a brief review of several prediction models in Section II. Section III narrates the proposed methodology, which includes input data collection, feature extraction, hyper-tuning of parameters, and the

classification approach employed. An experimental evaluation is carried out in Section IV. The overall summary is carried in Section V with future scope of directions.

2. Related Works

A comprehensive review of various traditional models related to the detection and prediction of SOH in EV batteries is presented here. A new hybrid scheme was suggested in [14], which combines a multi-head dilated temporal architecture of CNN with GRU for anticipating the levels of SOC. This model thus facilitates the simultaneous pattern learning over varied scales, thus allowing this technique to adopt new patterns more quickly. Moreover, the integration of explainable artificial intelligence models termed Shapley Additive exPlanations (SHAP) aims at achieving global interpretability for the prediction of SOC, thus offering precise quantification in the individual attribute influence. A comprehensive experiment was evaluated over varied temperature ranges and thus driving cycles to demonstrate the proposed scheme's effectiveness.

The Fractional-Order Method (FOM) assisted the online SOC and SOP estimation model for LIBs in EV, which was introduced in the work [15]. In order to identify two resistor consistent element phase networks that accurately describe the internal dynamics of a battery over a range of timeframes, the model parameters of the second-order FOM were first adjusted globally under a dynamic test stress profile. Partial Adaptive FOM (PA-FOM) was then developed to improve the model's SOC and SOP estimation performance. While an unscented Kalman filter-assisted iterative model was developed for predicting SOP, online SOC estimation was done using an adaptive extended Kalman filter model based on PA-FOM.

The online SOC and SOP co-estimation of LIBs in EVs was proposed in [16] using a Model Fusion Approach (MFM). The battery Open-Circuit Voltage (OCV)-SOC curve was first constructed using the Particle Swarm Optimization-Genetic Algorithm (PSO-GA) methodology in conjunction with two FOM RCCPE. This method depends solely on the dynamic load profile and does not require any prior information about the initial SOC. After identifying the parameter models, the Dual Extended Kalman Filter (DEKF) model based on the 1-RC scheme was used to estimate the battery's State Of Charge (SOC) using the extracted OCV-SOC curve. Additionally, two elements of battery polarization dynamics—current excitation and self-recovery—will be examined in the SOP window prediction.

In [17], a new machine learning model was proposed for reliable and accurate SOC prediction in EV batteries using the Differential Search Optimized (DSA) Random Forest Regression technique (RFR). Furthermore, a crucial problem that needs to be investigated is the exact choice of RFR architecture and hyperparameter integration. As a result, DSA

was used to find the best values for trees and leaves using the RFR technique. This DSA-optimized RFR strategy does not require a filter during the data preprocessing step and just requires sensors to monitor the battery's voltage and current, negating the requirement for in-depth knowledge of battery chemistry. According to an experimental result, the DSA-optimized RFR model achieves an RMSE value of 0.382% in the HPPC test using a LiNMC battery.

An online endwise state monitoring model was used in [18], depending on a multi-task transfer learning mechanism. This strategy was designed to enhance accuracy under different application scenarios. An experiment was conducted under varied working profiles, aging conditions, and temperatures to evaluate the model that covers a wide range of EV use. Comparing several benchmarks illustrates the superiority of the proposed scheme with enhanced computational efficiency and accuracy. Outcome reveals that MAE & RMSE of SOC and SOE estimation are lower than 2.31 & 3.31% respectively. A DL aided SOC prediction scheme was employed to ensure the representation of a reliable vector and sufficient extraction of features. So as to enhance the representation of battery data, an RNN-aided model was proposed. After that, aiming at fully extracted feature information, a multi-channel extended CNN model was presented to predict LIBs' SOC precisely. Depending on reliable vector representation and feature extraction, the suggested model offers enhanced performance of SOC prediction. The valuation shows that the suggested model was verified by a simulation test, which shows that the suggested scheme offers enhanced prediction performance with RNNs.

In [19], the Stochastic Model Predictive Control (SMPC) was proposed. For the energy management system strategy, a multiple linear regression of the engine and battery was initially created. Following that, a Markov chain-based velocity prediction model was created, considering the driving styles. Reference SOC will then be optimized through Dynamic Programming (DP), utilizing upcoming data. Finally, SMPC-assisted EMS and short-term optimum SOC will be formed. The results indicate that when radial basis function neural networks and backpropagation neural networks are compared, the Markov dependent model performs better in the prediction process.

An innovative solution was presented and compared a transformer model along with LSTM, Bi-directional LSTM, scheme, and Support Vector Regression (SVR). This model intends to offer new perspectives on the SOC of battery predictions using the BMW, NASA, Stanford University battery dataset, and real-time battery data attained from the L5 EV of the Musoshi brand, gathered for this model. The primary objective of this research was to employ a transformer model for real-time battery data, thus estimating them as an important stage in optimization and battery management of EV. A transformer model in this work attains an enhanced

outcome with an RMSE value closer to 1 (~0.99). The work presented in enhances the precision forecast and resilience regarding the remaining life of the battery. This model uses support vector networks and quantile regression to estimate the battery's health. Additionally, it integrates temporal convolutional networks and self-coding neural networks for processing and battery life data extraction, and lastly, it introduces a new prediction scheme. It was evident that the proposed scheme attains some benefits in estimating the lifespan of LIBs for EV. In addition, the findings of the model offer precise, quick, and flexible references to estimate the remaining life and condition of batteries.

An improved Multi-strategy improved Dung Dung Beetle Optimization model (MIDBO) was suggested in accurately predicting the SOC of electric load battery for optimizing the SOC prediction model of Extreme Learning Machine (ELM). Initially, PCA was employed to screen input features, thus reducing dimensionality. A dynamic spiral searching model was used in the breeding phase of the dung beetle approach, and the Levy flight model was incorporated at the foraging stage to escape the local optimum. Finally, adaptive t-distribution alteration and a dynamic selection strategy were employed to update the dung beetle position, thus improving convergence speed. This model enables the precise prediction of the electric loader's SOC battery. The experimental outcome shows a lower error rate for the proposed model.

The work suggested in presented a new hybrid neural network that integrates the GRU and LSTM schemes for SOH estimation. This model proves effective enhancement in the accuracy estimation of SOH and SOC with minimal training data needed. The primary contributions include the hybrid GRU-LSTM scheme, which enhances SOH/SOC accuracy, self-optimization capabilities, effective temperature variation handling without OCV-SOC lookup tables, and their application to various lithium battery types. An investigational outcome reveals that the proposed scheme offers RMSE 2% and MAE 1.7% for SOC, and RMSE 0.65% and MAE 0.85% for SOH.

2.1. Research Gap

A huge range of models, which includes DL models such as ConvLSTM, GAN, CNN, and hybrid schemes that combine various neural network models, was reviewed. From the survey, it was obvious that there is a need to enhance feature weighting and to enhance the accurate and robust creation of SOH estimation. Filling up these gaps could help create SOH estimation models, which would ultimately provide dependable and efficient BMSs for a variety of applications, including EV.

3. Proposed Work

The proposed working methodology based on DL aided SOH prediction is described in this section. Figure 2 depicts the entire flow of the proposed framework. Typically, the

SOH prediction model's workflow comprises subsequent stages like acquisition of data, selection of equivalent or relevant features, and estimation of SOH prediction. Real-time data collection from the particular dataset is the first step. The collected data will next go through a preprocessing step that includes data normalization and cleaning to get rid of any undesired noise issues. Following preprocessing, the data moves on to feature selection and classification phases. Complex, non-linear connections and patterns in time-series data, which are features of battery performance measures, can be handled by the DL scheme.

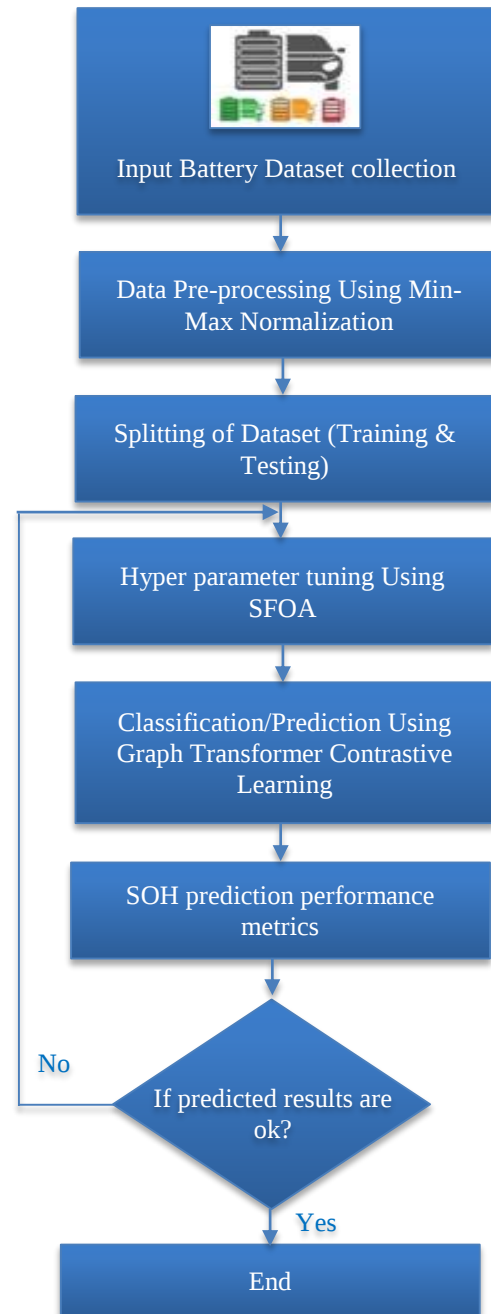


Fig. 2 Overall working flow of the proposed model

In contrast to conventional machine learning techniques, GNN, in particular, is more adept at identifying temporal correlations and intricately interacting with features, which results in a more accurate prediction of SOH. This capability is essential for creating reliable BMSs that can function well in a variety of scenarios with various battery types.

3.1. Input Battery Data Collection

The input battery dataset is gathered from Musoshi Company. Also, standard datasets are often employed in literature, like NASA [20], Stanford [21], and BMW i3 [22].

3.2. Preprocessing of Input Data

Preprocessing input data is a fundamental step before using any kind of data analysis or modeling technique. To guarantee the efficacy and caliber of analysis, this also entails transforming raw data into a clear and useful format. The data preprocessing steps used for LIB time series data used for SOH forecasting are data normalization and data purification. After cleaning, the raw data that depends on time series will be transformed into SOH-dependent data. It includes meticulously evaluating and correcting errors or inconsistencies that are present in raw data to enhance quality. Effective experimental data management is required to increase accuracy and DL scheme performance.

In order to provide battery data that displays periodic characteristics degradation, the data will first be cleaned by removing outliers and missing data, which are evaluated using intermediate data or moving averages. A dataset that relies on SOH lacks a comparable scale. One method commonly employed in the in-depth modeling approach to improve the convergence of the model and prediction accuracy is data normalization. A normalization approach is carried out using the min-max approach, which includes data scaling of the range 0 to 1. Hence, dataset normalization depending on SOH with the use of min-max normalization will be expressed using the subsequent equation:

$$x_n = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

In this, x_{min} denotes the minimum real data value, x_n symbolizes processed data, x represents original data, and the maximum value is signified by x_{max} . Moreover, this model facilitates simple and quick normalization of data, all the while staying within the desired range.

3.3. Hyperparameter Tuning by SFOA

The hyperparameters of the graph transformer contrastive learning framework are tuned by means of the SFOA approach [23, 24]. The Starfish Optimization Algorithm (SFOA) is a stochastic search optimization technique that may be inspired by the exploration, prey, and regeneration actions of starfish. Like previous metaheuristic approaches, SFOA has two stages: the exploration stage mimics the starfish's exploring

activities, while the exploitation step is built using models of prey and regeneration.

In the SFOA exploration stage, a hybrid search way based on the five arms of a starfish-among which are the eyes-is created by mixing five-dimensional and unidimensional search designs in order to construct the mathematical models.

The suggested search pattern is determined by the optimization problem's dimension D . When $D > 5$, it employs a five-dimensional search pattern; when $D \leq 5$, it employs a unidimensional search pattern. This enables it to overcome the limitations of the low efficiency of the vector search pattern and the sluggish convergence of the unidimensional search pattern.

The hunting and generation tactics produce SFOA during the exploitation stage. The primary updating technique during SFOA's exploitation phase is the preying method. It encourages the candidates to go toward better sites by using a concurrent two-directional search technique based on the information of two starfish.

Specifically, only the last starfish in the population ($i=N$) undergoes the regeneration stage, which is essential for the global convergence capability during the solving task. Lastly, during the exploration and exploitation stages, SFOA offers the same optimization opportunity. The details of the SFOA mathematical models are as follows:

3.3.1. Initialization Stage

During the SFOA initialization phase, starfish positions will be randomly generated across the design variable boundaries, which could be represented as a matrix, as illustrated:

$$X = \begin{bmatrix} X_{11} & X_{12} & \dots & X_{1D} \\ X_{21} & X_{22} & \dots & X_{2D} \\ \vdots & \vdots & \ddots & \vdots \\ X_{N1} & X_{N2} & \dots & X_{ND} \end{bmatrix}_{N \times D} \quad (2)$$

Here, X denotes a matrix for solving starfish positions having size $N \times D$, N denotes the size of the population, and D signifies the dimension of design variables. At the phase of initialization, every starfish position in the above equation will be computed by:

$$X_{ij} = l_j + r(u_j - l_j), \quad i = 1, 2, \dots, N, \quad j = 1, 2, \dots, D \quad (3)$$

Here, X_{ij} signifies the dimension of the j th location at the i th starfish, r signifies a random number between (0,1) and u_j and l_j signifies upper & lower limits of design variables at j th dimension, correspondingly. Once the position of the initialized matrix is generated, the fitness value of the entire

starfish is attained by evaluating the objective function, which could be memorized in the form of a vector as follows:

$$F = \begin{bmatrix} F(X_1) \\ F(X_2) \\ \vdots \\ F(X_N) \end{bmatrix}_{N \times 1} \quad (4)$$

In this, F denotes a matrix for storing and thus updating attained fitness value, having size $N \times 1$. Following initialization, SFOA moves into the main function and starts the stage of exploration and exploitation.

3.3.2. Exploration Stage

This method simulates the seeking capabilities of five arms of starfish with eyes embedded at the arm's end, as given in the image below, by establishing an exploration stage to replicate the exploratory behavior of starfish. New search patterns that combine a five-dimensional search pattern for $D > 5$ with a unidirectional search pattern for $D \leq 5$ at various optimization challenges are being explored. The five arms (or eyes) of a starfish will determine the threshold of the dimension.

The search space problem will be large if the optimization task is more than 5 ($D > 5$), requiring starfish to go over all five limbs in order to investigate their surroundings. Additionally, in order to guide their movement, starfish arms require the search agents' best position knowledge. Hence, the mathematical modeling of this stage is expressed as shown:

$$\begin{cases} Y_{i,p}^T = X_{i,p}^T + a_1(X_{best,p}^T - X_{i,p}^T)\cos\theta, & r \leq 0.5 \\ Y_{i,p}^T = X_{i,p}^T - a_1(X_{best,p}^T - X_{i,p}^T)\sin\theta, & r > 0.5 \end{cases} \quad (5)$$

In this, $Y_{i,p}^T$ & $X_{i,p}^T$ signifies the attained and present position of the starfish, correspondingly. $X_{best,p}^T$ signifies the p dimension of the present best position, p denotes five randomly chosen dimensions between dimensions D, $r \in (0,1)$. a_1 & θ will be computed by:

$$a_1 = (2r - 1)\pi \quad (6)$$

$$\theta = \frac{\pi}{2} \cdot \frac{T}{T_{max}} \quad (7)$$

Here, T is the present iteration, and T_{max} signifies maximum iteration. Cosine & sine term indicates that starfish arms might twist left or right for approaching foods have a similar probability. At the exploration stage, a_1 will be generated randomly to update positions at every candidate & iteration, and as the number of iterations increases, θ will change. At $\theta \in [0, \pi/2]$, two parameters may be used to assess the impact of the distance between the best location and the current location at a chosen renewing dimension. Equation (5) will use five-dimensional search patterns to update just five

dimensions of locations in order to ensure search ability and enhance search accuracy in the event that $D > 5$ at the optimization problem by comparing vectorial search patterns. After the position is changed outside of the design variable boundary, arms tend to stay in their old positions instead of shifting to the new ones. The aforementioned is mathematically expressed by:

$$X_{i,p}^{T+1} = \begin{cases} Y_{i,p}^T & l_{b,p} \leq Y_{i,p}^T \leq u_{b,p} \\ X_{i,p}^T & \text{Otherwise} \end{cases} \quad (8)$$

Here, p signifies the dimension updated, $l_{b,p}$ and $u_{b,p}$ signifies design variables' bounds, correspondingly. The exploration step uses unidimensional search patterns to update the location when the optimization task dimension is less than 5 ($D \leq 5$). In this situation, a starfish uses position information from other starfish to move one arm in search of a food source. A revised position is assessed by:

$$Y_{i,p}^T = E_t X_{i,p}^T + A_1(X_{k1,p}^T - X_{i,p}^T) + A_2(X_{k2,p}^T - X_{i,p}^T) \quad (9)$$

In this, $X_{k1,p}^T$ and $X_{k2,p}^T$ will be a p-dimensional position from two starfish selected randomly, correspondingly, A_1 and A_2 denotes two random numbers among $(-1, 1)$, p denotes a chosen number arbitrarily in D dimensions. E_t denotes starfish energy and is computed by:

$$E_t = \frac{T_{max} - T}{T_{max}} \cos\theta \quad (10)$$

Here, θ is computed from Equation (7). Similar to the current updating rule, starfish tend to remain in their former position rather than moving to the updated one if their reached position is outside of the border.

3.3.3. Exploitation Stage

Two update strategies will be devised at this stage in order to seek global solutions, taking into account preying and regeneration behaviors at the exploitation stage. SFOA employs a simultaneous two-directional searching technique for modeling the starfish prey stage, which requires the usage of information from other starfish and the current population's optimal position. Using a parallel two-directional search approach, five distances are first calculated between the optimal position and other starfish, and then two distances are randomly selected as acknowledgement for updating each starfish's position. A distance could be computed by:

$$d_m = (X_{best}^T - X_{m_p}^T), \quad m=1, \dots, 5 \quad (11)$$

In this, d_m are attained distances among global and other starfish, whereas m_p will be the five randomly chosen starfish. Therefore, updating the rule of every starfish's prey behavior will be modeled by:

$$Y_i^T = X_i^T + r_1 d_{m1} + r_2 d_{m2} \quad (12)$$

In this, r_1 and r_2 will be the random number among (0,1), and d_{m1} and d_{m2} will be randomly chosen at d_m . The figure below will show a starfish's feeding activity. The candidate's starfish will be traveling in the direction of the best guiding solution based on the parallel two-directional search method, while other candidates will have a comparable ability to mitigate local optima.

Furthermore, starfish are vulnerable to predation by other predators because of their slow mobility. In order to avoid being caught, starfish may cut themselves if they are caught by a predator. Consequently, the position and regeneration phase will be updated using:

$$Y_i^T = \exp(-T \times N / T_{max}) X_i^T \quad (13)$$

Here, T will be the current iteration, T_{max} denotes the maximum iterative number, and N denotes population size. Once the position attained from Equations (12) and (13) is out of its boundary design variables, the position will be set by:

$$X_i^{T+1} = \begin{cases} Y_i^T & l_b \leq Y_i^T \leq u_b \\ l_b & Y_i^T < l_b \\ u_b & Y_i^T > l_b \end{cases} \quad (14)$$

The regeneration stage will be required to prevent local solutions and improve global convergence, even if it contains a very minor number of function estimations (only calculating once during the exploitation stage of every epoch).

Algorithm 1: Starfish optimization algorithm for hyperparameter tuning

- Step 1: Initialize population
- Define the number of starfish agents (N)
 - randomly initialize N hyperparameter sets within defined bounds
 - Estimate each agent using the validation loss of the model
- Step 2: If the termination condition is not met (max iteration or convergence)
- Step 3: Divide the population into two groups
- Central starfish (found the best population)
 - Leg stars (remaining solutions)
- Step 4: Updation of central starfish (Exploitation-refinement of best solution)
- Apply local fine-tuning with the use of small perturbations
 - Ensure solution stays within bounds
- Step 5: Updation of leg stars (exploration-searching for a new solution)

- Adjust leg position with the use of mathematical functions (for instance, Levy flight)
- Balance among exploration (diversity) and exploitation (convergence)

Step 6: Regeneration stage (Handling of weak solution)

- Identify poorly performing solutions
- Replace them with new candidates' samples near stronger solutions

Step 7: Estimate fitness (model performance with new hyperparameters)

- Train the model using a new hyperparameter set
- Compute validation loss at the fitness score

Step 8: Return the best identified hyperparameter set

Henceforth, from this SFOA approach, hyperparameters are tuned accordingly, and the best fitness function is attained.

3.4. GNN-based Feature Extraction and Classification / Prediction using the Graph Transformer Contrastive Learning Framework

For the prediction purpose, the graph transformer contrastive learning framework is employed. A graph transformer contrastive learning framework integrates a GNN model with the transformer-based DL approach. The detailed description of this process is given in this section.

3.4.1. GNN Model for Extracting Features

A fusion process of GNN with the transformer model marks a pivotal growth in deep learning, with GNN performing better in the analysis of graph data and transformers advancing the task of sequencing. Existing models that dominate the sequencing process but fall under distributed computing include RNN, GRUs, and LSTM. The introduction of attention mechanisms, best shown by Google's BERT, transformed natural language processing by placing emphasis on pertinent data segments during processing.

In order to combine their strengths for improved job performance, the synergy between GNNs for local structure and Transformers meant for global dependencies is now being studied. The input of GNN is battery features (voltage, current, and temperature) with the following layers: batch_size, seq_len 3. In GCNConv (layer 1), the following are input and output (in=3, out = 64), followed by ReLU activation layer (in=64, out=128), global mean pooling layer, which averages across entire nodes, and output layer (batch_size, 128), and is the final embeddings of GNN. Most of the GNNs conform to message passing among neighbor nodes, which could be expressed by a subsequent iteration function:

$$m_v^{l+1} = f_{\theta}^l(h_v^l, \{h_u^l | u \in \mathcal{N}_v\}) \quad (23)$$

$$h_v^{l+1} = \sigma^l(g^l(h_v^l, m_v^{l+1})) \quad (24)$$

In this, f_θ^l , σ^l , g^l signifies a parametric function, that is, a neighborhood aggregation function, activation function, for instance, sigmoid and ReLU, with a combination function (mean, and summation) in the l -th layer, which passes on the graph. $\mathcal{N}_v$ signifies node v neighborhood & h_v^l signifies hidden embedding for v . This passing message in Equation (10) might repeat L times ($l \in \{1, 2, \dots, L\}$) till convergence. In this work, information might pass over the entire graph. On considering GCN, for instance, the function of message passing in GCN will be expressed as follows:

$$h_v^{l+1} = \sigma \left(W^l \sum_{u \in \mathcal{N}_v} \frac{h_u^l}{|\mathcal{N}_v|} + B^l h_v^l \right) \quad (25)$$

Here, W^l & B^l denotes a learnable parameter at the l -th layer.

Also, contrastive learning is employed with the utilization of a contrastive loss function, which aids in enhancing feature representations that are learned by the model. This, in turn, ensures that the same samples (battery cycles having the same SOH) will be placed closer in the feature space learned, whereas dissimilar ones (battery cycles having varied SOH values) will be pushed apart. The optimized parameters are shown in Table 1.

Table 1. Values of optimized parameter

Hyperparameter	Lower Bound	Upper Bound	Best Value
GCN Layers	2	5	3
Hidden Units per GCN Layer	32	256	128
Learning Rate for GNN	0.0001	0.01	0.001
Transformer Attention Heads	2	8	4
Transformer Dropout Rate	0.1	0.5	0.2
Contrastive Loss Margin	0.2	1.5	0.8
FC Layer Neurons	64	256	128

The GNN encoded features signify varied battery cycles, and the model is trying to group battery cycles that are similar together and thus separates dissimilar ones with the use of contrastive loss. It is specifically employed for modelling battery degradation, where cycles with the same health states must be clustered in the learned representation space. The contrastive learning computes the pairwise distance among GNN embeddings and thus applies contrastive loss (margin=0.5) function followed by a normalization layer and output shape (batch_size, 128). The contrastive loss function is defined as shown:

$$\mathcal{L} = (1 - \text{label}).d^2 + \text{label}.(\max(0, \text{margin} - d))^2 \quad (26)$$

At which d is the pairwise distance among two feature representations. Label 1 represents similar cycles (lower difference in SOH), and label 0 denotes dissimilar cycles (higher variation in SOH). Margin denotes hyperparameters that control the distance threshold.

3.4.2. Transformer Model

Deep Neural Networks (DNNs) have emerged as a key framework for prediction tasks in recent years, and the transformer model has developed significantly at the same time. This transformer model works well in a variety of situations where it can produce predictions for related activities all at once. In the field of machine learning, research on using transformers for prediction tasks becomes crucial.

Transformers, which were first used for jobs involving natural language processing, exhibit remarkable skills in recognizing intricate sequential patterns and managing long-term dependencies. Extending the usage of transformers beyond the traditional sequence-based application to include huge domains such as time series forecasting, classification of data or images, and financial predictions broadens their ability. This model delves into transformer use on predictive modeling, thus highlighting its strength for deciphering the relationship of complex data.

The transformer model used has encoder and decoder layers. The encoder layer acts as an input layer for transforming the input data. A positional encoding with cosine and sinusoidal functions will be added to encode the sequential information. Following this, four layers of encoder come into play, each consisting of multi-head self-attention and feedforward sub-layers. The encoder thus generates size vectors d_{model} . A model size signifies the vector space dimensionality at which the transformer model functions. The hyperparameter tuning identifies the hidden representation size and embeddings in the model. A decoder initiates with the input layer, thus utilizing data from the encoder. The input data is then transformed to a vector of size d_{model} .

Every layer of the decoder includes Multi-Head Self-Attention (MHSA) with feedforward sub-layers. Moreover, the decoder employs self-attention to the encoder output. An output layer thus aligns the process of prediction with the time series that are targeted. A prediction process relies on previous data points and thus uses a self-attention mechanism with the positional offsets. The transformer encoder has the following layers: input layer (batch_size, seq_len, 128), positional encoding layer, self-attention heads (multi-head=8, d_{model} =128) as the attention mechanism, feedforward network (hidden=256, and ReLU as 128), dropout layer (rate=0.1), normalization layer, and output layer (batch_size, seq_len, 128). The fully connected layer is the final prediction layer, and it comprises a linear layer (in=128, out=64), a final

linear layer for prediction (in=64, out=1), and an output layer (batch_size, 1).

Every encoder layer comprises self-attention and normalization and adds feedforward sublayers, with positional encoding applied to the input. A decoder layer thus incorporates an additional encoder-decoder model to facilitate the output and input sequence interaction. This model thus enables effective processing and contextual understanding of the sequential data. A transformer encoder model comprises identical stack layers. Every layer comprises two sub-layers: the first one is the MHSA mechanism, and the next one is the simple feedforward neural network. The residual connection is then used over each sub-layer, which is followed by the normalization layer. To enable the residual connection, all the sub-layers of the model, including embedding layers, are used, thus generating outputs with a fixed dimension of d_{model} . Likewise, the decoder consists of an identical stack layer. Every layer of the decoder consists of identical stack layers, and every layer of the decoder comprises sub-layers similar to the encoder. The third layer will then be introduced, which performs MHSA on the encoder's output. The residual links will be applied over each sub-layer, followed by a normalization layer similar to the encoder. This transformer model thus eliminates convolution and recurrence, relying instead on positional encoding to capture sequential data information. This positional encoding shares a similar dimensionality (d_{model}) as embeddings, thus allowing them to sum up. A positional encoding will be computed with the use of the following sine and cosine functions, as represented in the equation expressed below:

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10.000^{2i/d_{model}}}\right) \quad (15)$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10.000^{2i/d_{model}}}\right) \quad (16)$$

Here, pos signifies position, i signifies dimension, and d_{model} denotes model dimension. Every dimension of positional encoding will be signified as a sinusoidal function, at which the wavelength thus follows a geometric progression that ranges from 2π to $10,000 \cdot 2\pi$.

A Feed Forward Neural Network (FFN) in transformer comprises two linear transformations, with Rectified Linear Unit (ReLU) activation function employed among them. A simple but effective architecture model thus contributes to the model's expressive power. The function of FFN will be represented in the equation provided below:

$$FFN(H_0) = \text{ReLU}(H_0 W_1 + b_1) W_2 + b_2 \quad (17)$$

In this, H_0 signifies preceding layer output, whereas $W_1 \in \mathbb{R}^{d_{model} \times d_f}$, $W_2 \in \mathbb{R}^{d_{model} \times d_f}$, $b_1 \in \mathbb{R}^{d_f}$, and $b_2 \in \mathbb{R}^{d_{model}}$, a bias vector, which signifies trainable parameters. An input and output dimensionality are d_{model} , at which inner layer has dimensionality d_f .

The output of each preceding layer will be linked by residual connections to their subsequent input layers. These connections will be followed by a normalization layer, which is critical for stabilizing the process of training, thus enhancing convergence. The function of normalization is given in the equation below:

$$H_0 = \text{Layer Norm}(\text{Self attention}(X) + X) \quad (18)$$

$$H = \text{Layer Norm}(FFN(H_0) + H_0) \quad (19)$$

In this, self-attention is referred to as a module of self-attention, LayerNorm denotes a normalization function layer, and X denotes the layer's input. A core innovation of the architecture transformer lies in its self-attention mechanism, which is embedded in multi-head self-attention layers. The mechanism thus enables the scheme to estimate the relative significance of varied input tokens at the time of prediction. Not like existing models, which rely on fixed-size context windows, the mechanism of self-attention considers relationship simultaneously among entire input tokens.

A transformer uses the Query-key value model for computing the scaled dot-product attention, as expressed in the equation given below:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (20)$$

In this, queries $Q \in \mathbb{R}^{l_q \times d_k}$, the keys $K \in \mathbb{R}^{l_k \times d_k}$ having values $V \in \mathbb{R}^{l_v \times d_v}$ signifies core components in the self-attention mechanism. l_q and l_k signifies queries' length and values or keys correspondingly, whereas, d_k and d_v denotes key dimension (queries) & values.

Input values will be weighted by the attention score that is passed over multiple layers of transformer encoder blocks. An attention-weighted output vector will be processed over a fully connected feedforward network to generate the last prediction. An output of multiple attention heads, thus their function are run in parallel, and will be concatenated to produce the last output values of the multi-head attention model, as defined in the equation shown below:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(h_1, \dots, h_n) W^0 \quad (21)$$

$$\text{where } h_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \quad (22)$$

In this, W_i^Q , W_i^K , & W_i^V denotes parameter matrices.

This model uses a transformer DL model to predict SOH. A transformer is renowned for modeling sequential data efficiently, thus making it appropriate for the process of time series forecasting. A transformer model will be customized at the time of configuration to fit the dataset and the desired

requirements. This customization includes model architecture configuration, number of layers, attention model, and embedded dimensions for adapting the dataset and catering to desired needs. The optimization of hyperparameters is signified as a crucial step for improving transformer model performance, and for this, the Starfish Optimization Algorithm (SFOA) is employed. As of this proposed model, the final prediction of the SOH of the EV battery is carried out, and the performance estimation of this is carried out in the subsequent section.

4. Performance Analysis

The performance assessment is carried out for the proposed design, and the comparative analysis between the proposed and existing models is performed to validate the performance of the proposed design over others.

4.1. Dataset Description

The outcomes are attained using an EV dataset obtained from the Musoshi company. The standard datasets were also used, which are mostly employed in existing literature, including NASA, Stanford, and BMW i3. In the Musoshi EV dataset, the three-wheeled pop-up smaller vehicle offered by the Musoshi company has a total weight of 1100 kg. Based on the course design, the car will finish two circuits. A road map and SoC charge chart are shown in Figure 2. About 49 minutes will be needed by car to finish one lap. The car thus had a full charge once it began the initial lap and an 80% charge once it began the second one. A total of 194 data features, which include instantaneous current, voltage values, and current, were thus collected from the vehicle's BMS system. The fields of the Mushoshi dataset are provided in Table 2 [25]. NASA Ames Prognostics Center of Excellence (PCoE) has presented dedicated battery prognostic test platforms at which the NASA Li-ion battery Aging dataset portal (2023) was collected. At different temperatures, li-ion batteries run using three specific operating profiles: charging, discharging, and electrochemical impedance spectroscopy. Several current levels were employed for discharging till the battery voltage hits some thresholds. To cause a deep discharge aging effect, various thresholds will be lower than the OEM-recommended value of 2.7V. The aging of the battery is shown to be accelerated on repeated cycles of charging and discharging, and trials will be stopped after there is a drop in capacity of the battery by 30% (from 2Ah to 1.4Ah). Information has been prepared for the purpose of training.

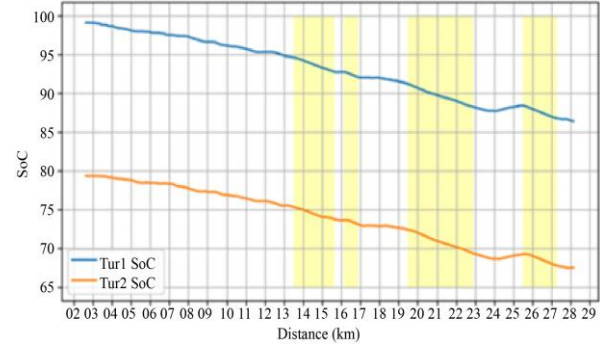
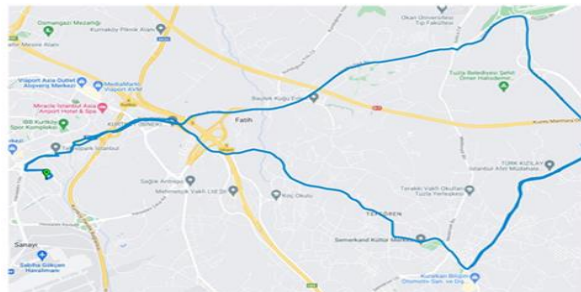


Fig. 3 Mushoshi vehicle routes for case tour 1 and tour 2 represented in the map, highlighting roads traversed in experiments

Figure 3 represents the Mushoshi vehicle routes for case tour 1 and tour 2, represented in a map, highlighting the roads traversed during experiments. A SOC profile corresponding to each tour will be plotted against travelled distance (in km). The blue line signifies the SOC profile for tour 1, whereas the orange line represents tour 2. A shaded region in the SOC plot resembles route sections at which significant alterations in energy consumption will be observed, potentially because of road conditions, variation of driving pattern, and traffic.

Table 2. Musoshi battery dataset

Symbol	Description
V_Cells	Total battery voltage (V)
T_Min	Minimum value of temperature read from battery temperature sensor ($^{\circ}\text{C}$)
T_Max	Maximum temperature value read from battery temperature sensor ($^{\circ}\text{C}$)
V_Avg	Average voltage value read from battery cells (mV)
SOH	Health Status
SOC_Trimmed	SOC value (%)
SOC_INTERNAL	More precise SOC value (%)
SOC	State of Charge
Pack_I_HALL	Current value (A)
V_MAX	Maximum voltage value read from battery cells (mV)
V_MIN	Minimum voltage value read from battery cells (mV)

In the Stanford dataset [21], both software and hardware are included. Arbin instruments LBT21024 were employed on the side of hardware. A system employs an Omega T – type thermocouple sensor and IncuMax IC – 500R thermal chamber from Arbin Instruments LBT21024.

Furthermore, Data Watcher & MITS Pro are employed on the software side. The comprehensive vehicle model, which includes the engine system and heating circuit, will be validated by the BMW i3 battery dataset that comprises 72 authentic driving tips that are recorded with the BMW i3 (60Ah) [22].

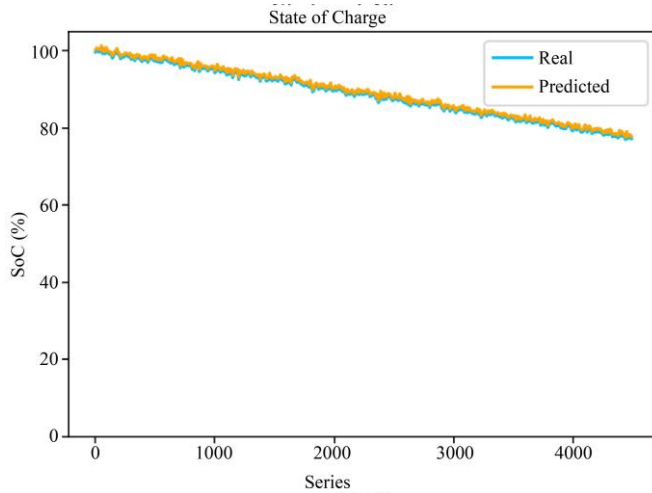


Fig. 4 Outcomes of the proposed GT-SOH-SFOA model to predict SOC on the Musoshi battery dataset

A proposed GT-SOH-SFOA model is trained to predict SoC using battery data. This scheme includes the determination of the relationship among the data points, thus adjusting the weights. The training thus happens to minimize the MSE loss function and measure how close the predictions are to the actual SoC values. Once training is completed, the performance of the model will be evaluated using test data. Metrics like R2 value, MAE, and RMSE will be employed to measure accuracy and prediction consistency. It is evident that both real and predicted values overlap one another, so the error rate is lower. The performance of this transformer model in predicting SOC is computed by various battery datasets, and the Musoshi battery dataset's computed value is shown in Figure 3. The performance of the GT-SOH-SFOA model in predicting the SoC is evaluated using various battery datasets, as shown in Figure 4.

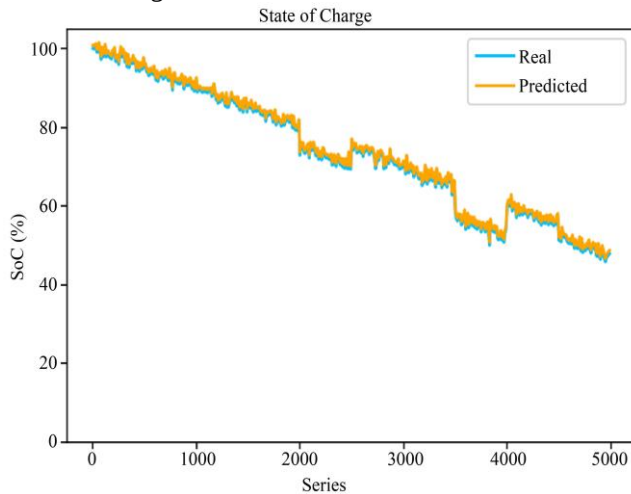


Fig. 5 Results of the GT-SOH-SFOA model for SoC prediction across the battery dataset, NASA

Figure 5 shows the results of the GT-SOH-SFOA model on predicting SOC over the battery dataset from NASA. The preprocessed data is restructured as input features and targeted

variables, after which training and testing models are separated. For instance, the NASA battery dataset B005 is employed for model training, and B006 is employed for testing purposes. A model is compiled and trained. With the use of training data from time series data, the model learned. A training process outcome is plotted, which illustrates how the loss function model was changed.

This model has made predictions on training data when compared to actual data. RMSE is computed for training data. An exact process for testing data is repeated, and RMSE will be computed. Moreover, MAE and R^2 scores assess the accuracy of the prediction model. Finally, the graph indicating the model's prediction and actual data will be plotted, as represented in Figure 4. It is evident that both real and predicted values overlap one another, and for this reason, the error rate is lower. The plot highlights the prediction consistency and any such deviation, thus showing the robustness of the model.

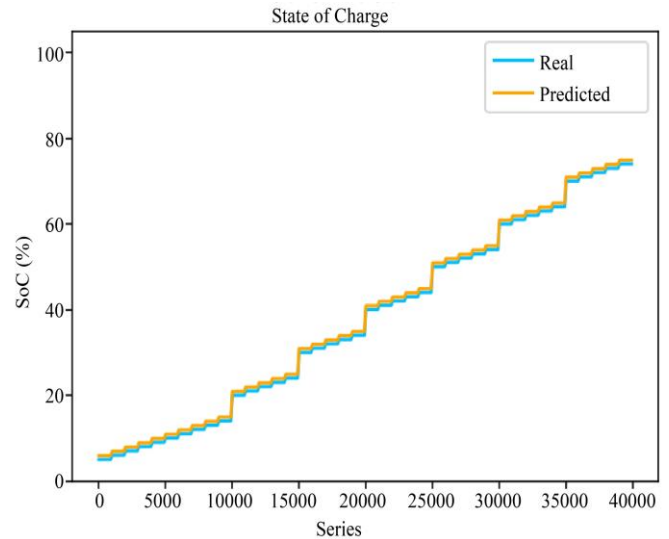


Fig. 6 Results of the GT-SOH-SFOA model for SoC prediction across the battery dataset_Stanford

Figure 6 shows the results of the GT-SOH-SFOA model on predicting SOC over the battery dataset from Stanford. A training process outcome is plotted, illustrating how the loss function model was changed. This model has made predictions on training data when compared to actual data. RMSE is computed for training data. An exact process for testing data is repeated, and RMSE will be computed.

Moreover, MAE and R^2 scores assess the accuracy of the prediction model. Finally, the graph indicating the model's prediction and actual data will be plotted, as represented in Figure 5. It is evident that both real and predicted values overlap one another, and for this reason, the error rate is lower. The plot highlights the prediction consistency and any such deviation, thus showing the robustness of the model.

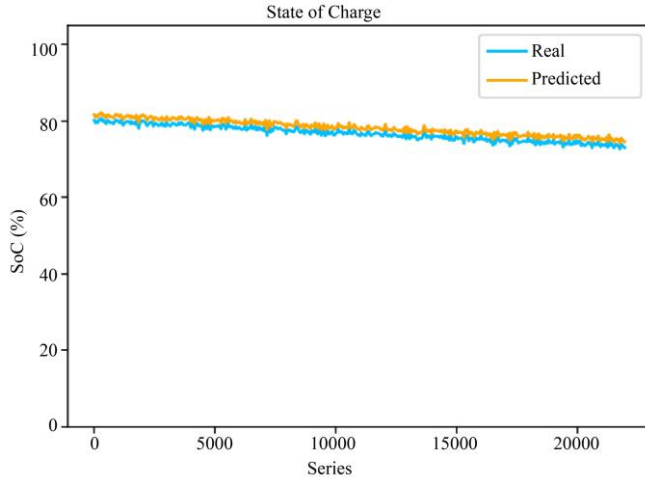


Fig. 7 Results of the Transformer model for SoC prediction across the battery dataset BMW i3.

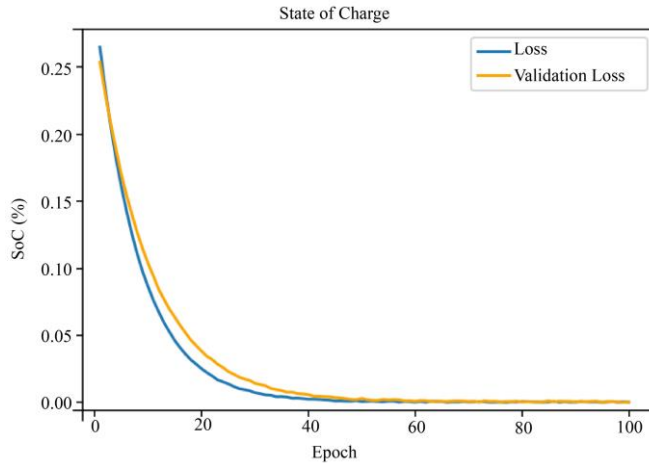


Fig. 8 Model loss results for the Musoshi battery dataset before optimization

Figure 7 shows the results of the transformer model on predicting SOC over the battery dataset BMW i3. A training process outcome is plotted, which illustrates how the loss function model was changed. This model has made predictions on training data when compared to actual data. RMSE is computed for training data. An exact process for

testing data is repeated, and RMSE will be computed. Moreover, MAE and R^2 scores assess the accuracy of the prediction model. Finally, the graph indicating the model's prediction and actual data will be plotted, as represented in Figure 6. The plot highlights the prediction consistency and any such deviation, thus showing the robustness of the model.

Figure 8 shows the model loss outcome for the Musoshi battery dataset before employing the SFOA optimization approach. The proposed model attains a lower loss rate compared to validation loss; however, without an optimization process, the loss will be slightly higher when comparing the outcomes attained after applying the optimization model, as shown in the subsequent outcome.

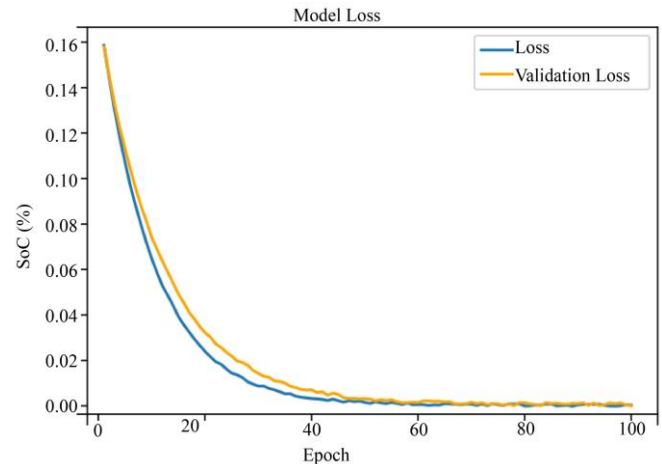


Fig. 9 Model loss results for the Musoshi battery dataset after optimization

Figure 9 shows the model loss outcome for the Musoshi battery dataset after employing the SFOA optimization approach. The proposed model attains a lower loss rate than validation loss, thus demonstrating their superiority in minimizing errors during training and validation after employing the SFOA model. A near-overlapping training and validation loss curves signify that the scheme is neither overfitting nor underfitting, thus ensuring generalizability.

Table 3. Results comparison for various performance metrics

Dataset	Metrics	ELM	TCN-SVN	LSTM+GRU	TCRN	RNN	GT-SoH-SFOA
NASA	RMSE	2.927	3.437	17.873	19.500	18.900	2.083
	MAE	2.885	3.358	17.507	18.800	18.400	2.010
	R^2 Score	2.209	2.209	1.572	1.400	1.450	2.205
	Max Error	8.456	8.169	9.456	10.200	9.800	7.908
Stanford	RMSE	3.235	3.132	6.628	7.500	7.000	3.132
	MAE	3.085	2.942	4.509	5.300	4.900	2.764
	R^2 Score	2.210	2.210	2.042	1.900	1.950	2.206

	Max Error	7.016	6.950	7.072	7.900	7.500	7.113
BMW i3	RMSE	1.914	1.795	4.988	5.500	5.200	1.500
	MAE	1.548	1.558	4.830	5.100	4.900	1.470
	R ² Score	2.209	2.209	1.495	1.300	1.350	2.210
	Max Error	2.421	1.955	4.908	5.600	5.400	2.421
Musoshi - L5	RMSE	2.381	2.155	4.488	5.000	4.700	1.377
	MAE	2.217	1.967	4.261	4.800	4.500	1.343
	R ² Score	2.173	2.185	1.555	1.400	1.450	2.208
	Max Error	3.825	3.536	4.769	5.200	5.000	3.346

Table 3 shows the performance assessment of various metrics evaluated for proposed and various existing models like ELM [26], TCN-SVN [27], LSTM-GRU [28], TCRN [29], and RNN [30] for four kinds of battery datasets like Musoshi, NASA, Stanford, and BMW i3. The evaluation of the outcomes as per the datasets reveals that the transformer model consistently attains low max error for the entire dataset, thus indicating robustness in minimizing extreme variations when comparing other existing models. This, in turn, proves the superiority of the proposed model over other existing schemes.

5. Conclusion

In this paper, a GT-SoH framework was proposed for effectively addressing challenges of precise SOH prediction for EV batteries by integrating GNNs, Transformer-based temporal modeling, contrastive self-supervised learning, and Starfish Optimization (SFOA) for hyperparameter tuning. On leveraging GNNs, this model efficiently captures spatial dependencies among battery cells, whereas the transformer

encoder model has long-range temporal degradation patterns. The integration of the contrastive learning function in the GNN model improves the generalization ability on learning robust representations of features from the unlabeled battery datasets. In addition, SFOA optimizes hyperparameters, thus ensuring a balanced trade-off between exploitation and exploration. An experimental estimation on the benchmark battery dataset illustrates that the proposed GT-SOH-SFOA outperforms existing schemes significantly in terms of prediction, robustness, and generalization over varied battery conditions. A hybrid loss function thus enhances the performance of the model on mitigating the overfitting issue, thereby ensuring precise estimation of SOH. A presented framework offers a scalable, interpretable, and real-time solution for monitoring battery health, thus paving the way to improve battery safety, longevity, and management of optimal energy in the next-generation EV battery system. Future research might explore the extension of this model to multi-modal battery datasets and real-world deployment in Battery Management Systems (BMS).

References

- [1] C. Pratheeba, and Sukumar, "Hybrid HGRN-SCSO Technique for Enhanced Prediction of Remaining Useful Life in EV Batteries," *Electrical Engineering*, vol. 107, pp. 5311-5323, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [2] Vankamamidi S. Naresh et al., "Optimizing Electric Vehicle Battery Health Monitoring: A Resilient Ensemble Learning Approach for State-Of-Health Prediction," *Sustainable Energy, Grids and Networks*, vol. 42, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [3] Pooja Kumari, and Niranjana Kumar, "Hybrid Optimized Deep Learning Approach for Prediction of Battery State of Charge, State of Health and State of Temperature," *Electrical Engineering*, vol. 106, pp. 1283-1290, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [4] Ramprabu Jayaraman, and Rani Thottungal, "Accurate State of Charge Prediction for Lithium-Ion Batteries in Electric Vehicles using Deep Learning and Dimensionality Reduction," *Electrical Engineering*, vol. 106, pp. 2175-2195, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [5] Mona Faraji Niri et al., "State of Power Prediction for Lithium-Ion Batteries in Electric Vehicles via Wavelet-Markov Load Analysis," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 9, pp. 5833-5848, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [6] Khalid Akbar et al., "A Machine Learning-Based Robust State of Health (SOH) Prediction Model for Electric Vehicle Batteries," *Electronics*, vol. 11, no. 8, pp. 1-14, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [7] Guangzhong Dong, Weiji Han, and Yujie Wang, "Dynamic Bayesian Network-Based Lithium-Ion Battery Health Prognosis for Electric Vehicles," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 11, pp. 10949-10958, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]

- [8] Guangyi Yang et al., “State of Charge Estimation of Lithium-Ion Battery for Underwater Vehicles Using MM-UKF Under Hierarchical Temperature Compensation,” *IEEE Access*, vol. 12, pp. 95831-93845, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [9] Xiangbo Cui, and Teta Hu, “State of Health Diagnosis and Remaining Useful Life Prediction for Lithium-ion Battery Based on Data Model Fusion Method,” *IEEE Access*, vol. 8, pp. 207298-207307, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [10] Jingyun Xu et al., “State of Health Diagnosis and Remaining Useful Life Prediction of Lithium-Ion Batteries based on Multi-Feature Data and Mechanism Fusion,” *IEEE Access*, vol. 9, pp. 85431-85441, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [11] Geunsu Kim et al., “Electric Vehicle Battery State of Charge Prediction based on Graph Convolutional Network,” *International Journal of Automotive Technology*, vol. 24, pp. 1519-1530, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [12] Jianqiang Gong et al., “Predictive Modeling for Electric Vehicle Battery State of Health: A Comprehensive Literature Review,” *Energies*, vol. 18, no. 2, pp. 1-37, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [13] Chi Zhang et al., “Novel Battery State of Health Estimation and Lifetime Prediction Method based on the Catboost Model,” *Energy & Fuels*, vol. 38, no. 11, pp. 10310-10323, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [14] Anantha Padmanabhan N K et al., “An Interpretable Electric Vehicles Battery State of Charge Estimation using MHDTCN-GRU,” *IEEE Transactions on Vehicular Technology*, vol. 73, no. 12, pp. 18527-18538, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [15] Ruohan Guo, and Weixiang Shen, “Lithium-ion Battery State of Charge and State of Power Estimation based on a Partial-Adaptive Fractional-Order Model in Electric Vehicles,” *IEEE Transactions on Industrial Electronics*, vol. 70, no. 10, pp. 10123-10133, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [16] Ruohan Guo, and Weixiang Shen, “A Model Fusion Method for Online State of Charge and State of Power Co-Estimation of Lithium-Ion Batteries in Electric Vehicles,” *IEEE Transactions on Vehicular Technology*, vol. 71, no. 11, pp. 11515-11525, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [17] M.S. Hossain Lipu et al., “Real-Time State of Charge Estimation of Lithium-Ion Batteries using Optimized Random Forest Regression Algorithm,” *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 1, pp. 639-648, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [18] Yunhong Che et al., “Battery States Monitoring for Electric Vehicles based on Transferred Multi-Task Learning,” *IEEE Transactions on Vehicular Technology*, vol. 72, no. 8, pp. 10037-10047, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [19] Xinyou Lin et al., “Stochastic Model Predictive Control Strategy With Short-Term Forecast Optimal SOC for a Plug-In Hybrid Electric Vehicle,” *IEEE Transactions on Transportation Electrification*, vol. 1, no. 4, pp. 8685-8697, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [20] Li-ion Battery Aging Datasets, NASA Open Data Portal, 2023. [Online]. Available: <https://data.nasa.gov/dataset/li-ion-battery-aging-datasets>
- [21] Edoardo Catenaro, and Simona Onori, “Experimental Data of Lithium-Ion Batteries Under Galvanostatic Discharge Tests at Different Rates and Temperatures of Operation,” *Data in Brief*, vol. 35, pp. 1-8, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [22] Matthias Steinstraeter, Johannes Buberger, and Dimitar Trifonov, “Battery and Heating Data in Real Driving Cycles,” *IEEE Dataport*, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [23] V.P. Kavitha et al., “An Efficient Modified Black Widow Optimized Node Localization in Wireless Sensor Network,” *IETE Journal of Research*, vol. 70, no. 12, pp. 8404-8413, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [24] Changting Zhong et al., “Starfish Optimization Algorithm (SFOA): A Bio-Inspired Metaheuristic Algorithm for Global Optimization Compared with 100 Optimizers,” *Neural Computing and Applications*, vol. 37, pp. 3641-3683, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [25] Metin Yılmaz, Eyüp Çınar, and Ahmet Yazıcı, “A Transformer-Based Model for State of Charge Estimation of Electrical Vehicle Batteries,” *IEEE Access*, vol. 13, pp. 33035-33048, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [26] Wei Ding et al., “State of Charge Prediction for Electric Loader Battery based on Extreme Learning Machine,” *IEEE Access*, vol. 13, pp. 3696-3706, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [27] Xinyue Li, and Jiangwei Chu, “Lithium Battery Life Prediction for Electric Vehicles using Enhanced TCN and SVN Quantile Regression,” *IEEE Access*, vol. 13, pp. 12581-12595, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [28] Saeid Jorkesh et al., “Battery State of Charge and State of Health Estimation Using a New Hybrid Deep Neural Network Approach,” *IEEE Access*, vol. 13, pp. 12566-1258, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [29] Juan Wang et al., “Temporal Convolutional Recombinant Network: A Novel Method for SOC Estimation and Prediction in Electric Vehicles,” *IEEE Access*, vol. 12, pp. 128326-128337, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [30] Fen Zhao et al., “Lithium-ion Batteries State of Charge Prediction of Electric Vehicles using RNNs-CNNs Neural Networks,” *IEEE Access*, vol. 8, pp. 98168-98180, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]