processes

MDPI

Article

Discharge Voltage Prediction Model of Batteries in Different Degradation States Based on IWOA-ATCN

Jingwei Yang *, Yitong Chen, Qiang Huang, Guilong Wu, Lin Liu, Zhimin Yang and Yu Huang

Power Dispatching & Control Center of China Southern Power Grid, Guangzhou 510525, China
* Correspondence: yangjw2@csg.cn

Abstract: As a key technology for future decarbonization, storage batteries are widely used in areas such as electric vehicles and power systems. However, battery aging inevitably limits their broader application. To address the low accuracy in predicting discharge voltage under different aging states, this paper proposes the IWOA-ATCN method based on a TCN model with a sliding window mechanism. First, the improved whale optimization algorithm (IWOA) is employed to optimize the hyperparameters in the TCN model, including window size and sliding step, to obtain the optimal sample structure. Then, the temporal attention mechanism is introduced into the TCN model to accurately capture the temporal correlation of discharge voltage, thereby improving the prediction accuracy of long time series data. Finally, the model is tested on the NASA dataset with an RMSE of 0.0072, MAE of 0.0046, and R^2 of 0.9984. The test results on the PL Sample dataset showed RMSE of 0.0081, MAE of 0.0040, and R^2 of 0.9983. It is indicated that the prediction accuracy and stability of the IWOA-ATCN model are higher than other models, such as BP, RNN, and LSTM.

Keywords: storage battery; voltage prediction; degraded state; whale optimization algorithm; TCN

check for updates

Academic Editors: Massimo Caruso and Qunjie Xu

Received: 15 November 2024
Revised: 12 December 2024
Accepted: 19 December 2024
Published: 28 December 2024

Citation: Yang, J.; Chen, Y.; Huang, Q.; Wu, G.; Liu, L.; Yang, Z.; Huang, Y. Discharge Voltage Prediction Model of Batteries in Different Degradation States Based on IWOA-ATCN. *Processes* **2025**, *13*, 46. https://doi.org/10.3390/pr13010046

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the transition and upgrading of the energy structure, energy storage systems provide an important technical guarantee for the stable power supply of the grid [1]. Due to the high mass and volume energy intensity, long cycle lifespan, high safety, and low self-discharge rate of storage batteries [2], they play an increasingly important role in the power system. However, in the actual use process, storage batteries will experience different degrees of degradation, which is manifested in the reduction of battery capacity and service life [3,4]. To ensure the security and efficient operation of storage batteries, they must be comprehensively monitored and controlled. On the one hand, battery management systems (BMS) assess the state of charge (SOC) of batteries through real-time monitoring of important parameters, such as voltage, current, and temperature, to avoid insufficient power supply or other safety hazards due to battery degradation in the service process [5]. On the other hand, due to degradation, electrical failure, or wrong operation of batteries, they may malfunction and manifest as abnormal voltage [6]. Overcharging of the battery or the failure of charging circuit protection may cause overvoltage, and internal discharge or short-circuiting of the battery may cause undervoltage. Therefore, it is essential to accurately predict the voltage of a running battery to assess the state of charge and prevent battery failure.

The degradation of batteries during use reduces their operational reliability, so assessing the state of charge of batteries is an important function of concern for BMSs [7]. To assess the charge state of a battery, it is usually required to estimate the maximum capacity of the battery through discharge tests, a process that involves analyzing the voltage variation curves and relies on a complex testing process, heavy measuring equipment, and a strict application environment [8]. The discharge test is not only time-consuming but also requires several repetitive operations, which significantly increases the workload and labor cost [9]. A large number of research works have been devoted to SOC assessment through capacity prediction to reduce discharge testing time, which is classified into model-based and data-driven approaches [10,11]. Model-based methods build dynamic models to simulate the physiological/electrochemical behaviors of the battery and estimate the decay condition of the battery through closed-loop control and real-time parameter update filters [12]. The effectiveness of model-based approaches is highly dependent on the accuracy of modeling [13]. However, accurate physical modeling is difficult due to the complexity of the degradation process itself and the multiple unpredictable sources that lead to its development [14], which may affect the accuracy of the model's predictions. Inaccurate predictions may lead to overcharging or deep discharging, which can shorten battery life and efficiency and have an impact on battery management systems and the performance of devices using batteries [15]. In recent years, battery condition monitoring has mostly been achieved through data-driven methods, as shown in Table 1.

Table 1. Related works on battery state assessment and prediction.

Reference	Year	Types of Batteries	Research Content	Method	Algorithm	Data Set
[1]	2018	lithium-ion	SOE and Voltage Response Prediction	model-driven	Kalman Filter	Laboratory test data
[16]	2019	lithium-ion	Capacity attenuation prediction	model-driven	Gaussian process	NASA
[17]	2020	LiFePO4	Voltage and power prediction	model-driven	/	Laboratory test data
[18]	2021	lithium-ion	Voltage prediction and state estimation	data-driven	FNN	Commercial battery data
[19]	2024	lithium-ion	SOH estimation	data-driven	BI-GRU	NASA
[20]	2020	lithium-ion	SOH estimation	data-driven	GRU-CNN	NASA and Oxford Battery Degradation
[6]	2024	lithium-ion	Voltage anomaly prediction	data-driven	informer	Laboratory test data
[21]	2023	lithium-ion	Voltage prediction	data-driven	Dynaformer	Laboratory test data

The data-driven method uses huge amounts of historical data and machine learning algorithms to build a voltage profile prediction model, which can effectively shorten the discharging time and improve the overall efficiency of battery management. For example, a model based on the decision tree algorithm and battery discharge data has been developed to predict voltage and temperature [22]. The LSTM neural network is used for multiple forward-step voltage predictions and is combined with alarm thresholds to achieve fault diagnosis [23]. Scholars have studied the voltage prediction accuracy of eight machine learning methods, among which the neural network model has the highest accuracy [24]. Among them, combining precise battery dynamic modeling with recursive neural networks results in a prediction error of less than 5%, but precise battery dynamic modeling is required [25]. Although the above work contributes to the assessment of battery degradation state through voltage prediction, the degradation of batteries depends on the cycling

conditions [26]. Most of the current methods assume that the future cycling conditions are fixed [27], which makes it difficult to ensure the accuracy of voltage predictions for batteries with varying degradation characteristics.

Discharge voltage prediction tasks usually need to deal with long time series, while conventional models, such as LSTM and RNN, cannot handle time series data in parallel [6]. As the length of time series data increases, performance degrades in handling long-term dependency relationships. The temporal convolution network (TCN) was proposed in 2018 by Bai et al. It utilizes dilated convolution to maintain a longer receptive field [28], and its model design architecture is relatively simple, mainly relying on convolutional layers and residual connections, thus simplifying model design. LSTM and RNN feature complex gating mechanisms, resulting in higher computational overhead [29]. TCN has gradually been applied to urban expressway traffic flow [30], stock price [31], precipitation [32], as well as other time series prediction tasks. Empirical studies have shown that TCN is generally superior to RNN and LSTM in tasks involving time series prediction and modeling, especially when data have long-term dependencies or complex time patterns [33].

Therefore, this paper constructs a storage battery discharge voltage prediction model based on IWOA-ATCN for different aging states, which includes an improved whale optimization algorithm and a temporal convolutional network that incorporates an attention mechanism, allowing the model to learn the trends in battery discharge voltage curves across different degradation states. The main contributions of this paper are as follows:

1. A TCN model that introduces an attention mechanism is constructed. Considering that the fully connected layer in TCN will ignore the correlation between different positions in the voltage curve, the temporal attention mechanism was introduced to allocate different weights based on the relevance of the time series data, allowing for more accurate capture of the data correlations.
2. An improved whale optimization algorithm (IWOA) is proposed to realize the hyperparameter optimization search of the TCN model and to enhance the generalization of the model for storage batteries with different degradation states. To address the issues of WOA easily falling into local optimal solutions and converging slowly, the Tent mapping initialization method is used to generate the initial population positions, and a nonlinear convergence factor is introduced to balance the trade-off between global and local search and improve the search efficiency and accuracy of the algorithm. Inertia weights are introduced to keep each individual in the previous direction during the iteration process, allowing the individual to be influenced by the combined effect of the historical optimal solution and its current individual position.
3. The IWOA-ATCN-based discharge voltage prediction model for storage batteries with different degradation states is proposed. The IWOA algorithm is used to optimize the two hyperparameters of the sliding window and step size of the ATCN-based battery discharge voltage model, which improves the prediction accuracy of the prediction model and the generalization of the discharge curves for batteries with different degradation states.

2. Relevant Theories

To solve the battery voltage discharge prediction problem, this paper proposes an IWOA and ATCN based on the temporal attention mechanism. To effectively capture the important information and long-time dependencies in the time series, this paper uses the IWOA algorithm to simulate whale predation behavior, which optimizes two hyperparameters, the sliding window and the step size of the battery discharge voltage prediction model in the research space, and the optimal parameters are used to train the TCN prediction model with a temporal attention-based mechanism for voltage prediction.

2.1. Principle of Optimization Algorithm IWOA

The Whale Optimization Algorithm (WOA) is a population-intelligent optimization algorithm that simulates the food-seeking behavior of whales [34]. The algorithm is divided into three phases, namely encirclement, attacking, and searching, dynamically adjusting the position of each whale to maximize the optimization efficiency and obtain the globally optimal solution.

In the encirclement phase, the position of the optimal whale $X_{\text{best}}(t)$ is assumed to be at a distance D from the prey, as shown in Formula (1). Other whales will gradually approach the optimal whale to encircle the prey while simultaneously using Formula (2) to adjust and update their position $X(t+1)$:

$$D = |C \cdot X_{best}(t) - X(t)| \tag{1}$$

$$X(t+1) = X_{best}(t) - A \cdot D \tag{2}$$

where t is the number of iterations and the coefficient vectors A and C are calculated by the following equation:

$$\begin{aligned} A &= 2a \cdot r - a \\ C &= 2r \end{aligned} \tag{3}$$

where r is a vector of random numbers with values ranging from 0 to 1, and a is a convergence factor whose value is reduced from 2 to 0 after continuous iterations, and the expression for a is

$$a = 2 \times (1 - \frac{t}{T_{\max}}) \tag{4}$$

where $T_{\max}$ is the maximum number of iterations.

The attack phase consists of two strategic mechanisms: contraction and spiral. The contraction mechanism updates the position of the whale using Formula (2); the spiral mechanism mimics the whale's hunting method by spitting bubbles to calculate the distance D' from the whale to the prey and update the whale's position, as in Formula (5):

$$\begin{aligned} D' &= |X_{best}(t) - X(t)| \\ X(t+1) &= D' \cdot e^{bl} \cos(2\pi l) + X_{best}(t) \end{aligned} \tag{5}$$

where b is a constant and l is a random number in the range $[-1,1]$. Both strategies have a trigger probability of 0.5, and they form the mathematical model of the attack phase.

In the search phase, to avoid local optimization, the whale performs a global search when $|A| \geq 1$ as follows [35]:

$$D = |C \cdot X_{rand}(t) - X(t)| \tag{6}$$

$$X(t+1) = X_{rand}(t) - A \cdot D \tag{7}$$

where $X_{rand}(t)$ is the current randomly selected whale position.

To overcome the algorithm's tendency to easily fall into local optima and its slow convergence, this paper improves the population initialization process, the convergence process, and the attack phase. The initialization phase of WOA usually adopts a random initialization method; the points generated by random initialization are uniformly distributed in the two-dimensional plane, which may result in a longer research time to achieve better optimization results. To enhance the search efficiency, this paper employs the Tent mapping initialization method, which can generate population positions with

randomness and fractal characteristics to enhance diversity and local search ability. The mathematical formula of Tent mapping is as follows:

$$x_{m+1} = \begin{cases} 2x_m, x_m < 0.5 \\ 2(1 - x_m), x_m \geq 0.5 \end{cases} \tag{8}$$

where x_0 takes a random value, and m represents the number of the whale. The locations of both initialized whale populations are shown in Figure 1, with a denser distribution of points in the Tent mapping method near the boundary region and a sparser distribution in the central region.

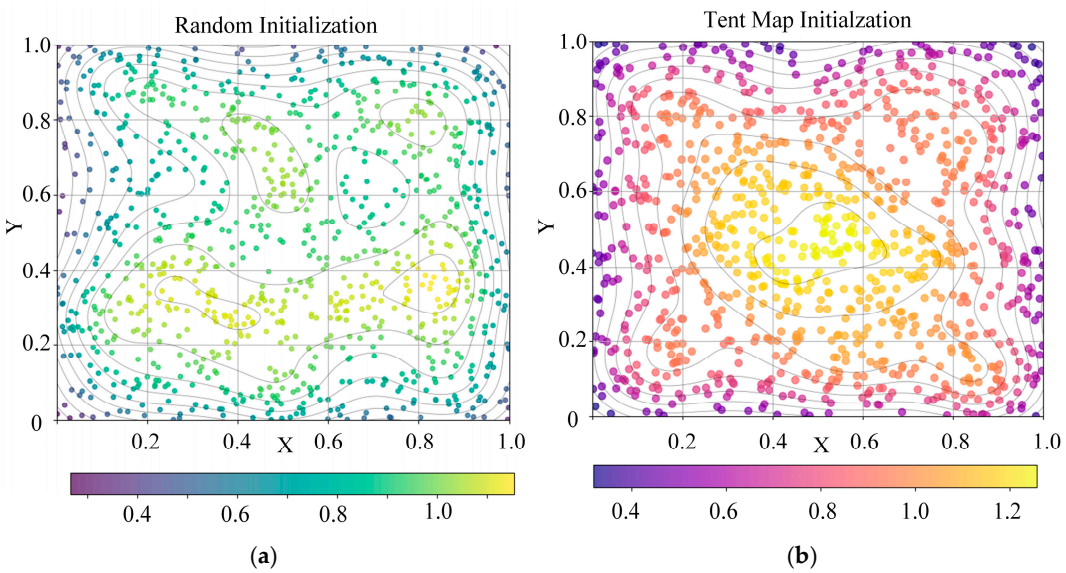


Figure 1. (**a**) Random initialization location map; (**b**) tent initialization location map.

The convergence factor a decreases in a linear manner with the iteration number; in order to improve the convergence speed of the algorithm, the nonlinear decreasing formula is designed, and the linear convergence factor is changed to a nonlinear convergence factor, which is shown in Formula (9). The introduction of the power exponent n makes the decreasing rate of a more flexible and adjustable and enhances the initial exploration and later convergence.

$$a = 2 \times (1 - (\frac{t}{T_{\max}})^{n}) \tag{9}$$

where n is the factor that controls the nonlinearity. By adjusting the value of n, it is possible to change the strength of the converged nonlinearity. When n = 1, it degenerates to its original linear form; when n > 1, the convergence process slows down, and the exploration phase becomes longer; when n < 1, the convergence process speeds up, the exploration phase becomes shorter, and the exploitation phase starts earlier.

In the attack phase, the introduction of inertia weights $\omega(t)$ keeps each individual moving in its previous direction during iteration and reduces oscillations during individual updates, thus accelerating convergence. The formula for the inertia weights is shown in (10).

$$\omega(t) = \omega_{initial} - (\omega_{initial} - \omega_{final}) \cos\left(\frac{\pi}{2}\left(1 - \frac{t}{T_{\max}}\right)\right) \tag{10}$$

where $\omega_{initial}$ is the initial value of inertia weights, and ω_{final} is the final value of inertia weights. The introduction of inertia weights better balances the algorithm's global and

local search capabilities, and the expressions for the attack phases after the introduction of inertia weights are as follows:

$$X(t+1) = \begin{cases} X_{best}(t) - \omega(t)A \cdot D, p < 0.5 \\ \omega(t)D' \cdot e^{bl} \cdot \cos(2\pi l) + X_{best}(t), p \geq 0.5 \end{cases} \quad (11)$$

where p is a random value in the range [0,1].

The flow of IWOA algorithm proposed in this paper is as follows (Algorithm 1).

Algorithm 1. The IWOA Algorithm

Input: Maximum number of iterations ($T_{\max}$), The number of whales (N), Dimension of the problem (dim), Initial population random value (x_0), Power factor of nonlinear convergence factor (n), Initial inertia weight ($\omega_{initial}$), Final inertia weight (ω_{final})
Output: Global Optimizer (x_{best}), The objective function value corresponding to the global optimal solution(f_{best})

for i from 1 to N **do**//Initializing whale position
    $x[i]$ = TentMapping($x_o[i]$, dim)//Using tent mapping
end for
for i from 1 to N//Set initial optimal solution
    x_{best}= argmin($f(x[i])$)
f_{best}= $f(x_{best})$//$f(x)$ is the objective function used to evaluate the position of whales
end for
for t from 1 to $T_{\max}$ **do**
    a $= 2 \times (1 - (t/T_{\max})^n)$//Calculate convergence factor
$\omega(t) = \omega_{initial} - (\omega_{initial} - \omega_{final}) \cos(\frac{\pi}{2}(1 - t/T_{max}))$//Calculating inertia weight
    for i from 1 to N **do**
        $A = 2 \times r - 1$//Where, r = random(0,1)
        $C = 2 \times r$
        $D = |x_{best} - x[i]|$//Calculating the distance between the current whale and the optimal whale
            if $D <$ threshold **then**//threshold is the set distance threshold
                $x[i]$ = x[$random_index$] + random(0,1) $\times$ (x_{best}− x[$random_index$])//Performing a global search
            else
                if random(0,1) < 0.5 **then**//Encirclement and capture
                    $x[i]$= $x[i]$+A $\times$ $|C \cdot x_{best} - x[i]|$//Using contraction mechanism
                else
                    $x[i]$= x_{best}+$b \cdot sin(l) \cdot |x_{best} - x[i]|$//Using a spiral a mechanism, where, l = random(−1,1)
                end if
            end if
            if $f(x[i]) < f_{best}$ **then**//Update the optimal solution
                x_{best}= $x[i]$
                f_{best}= $f(x_{best})$

Algorithm 1. Cont.

```
            end if
        end for
    end for
    return x_best, f_best
```

2.2. Principle of ATCN

TCN is a deep learning model comprised of dilated causal convolution structures and residual connections, specifically designed for modeling time series data. It has the ability to capture long-range dependent relationships. In the case of time series data, temporal order is critical because the predicted values depend entirely on the previous data. Consequently, the TCN modifies the propagation of context information found in classical convolutional networks to transmit only past information by utilizing causal convolution modules. The formula for calculating the conditional probability of causal convolution is presented in Formula (12).

$$p(x) = \prod_{i=1}^{t} p(x_t \mid x_1, \cdots x_t) \tag{12}$$

where $x_1, \ldots, x_t$ represents the time series data.

The stacking of causal convolutional layers ensures that temporal order is maintained in the data of the time series. Nevertheless, as the time series data length increases, simply stacking such convolution layers requires a sufficient number of layers to meet the historical memory length requirements. To address this, the TCN model introduces dilated convolution modules. Dilated convolution maintains the information density of time series data in the convolution operation by controlling the stride of the convolutional kernel. The structure of dilated causal convolution is illustrated in Figure 2.

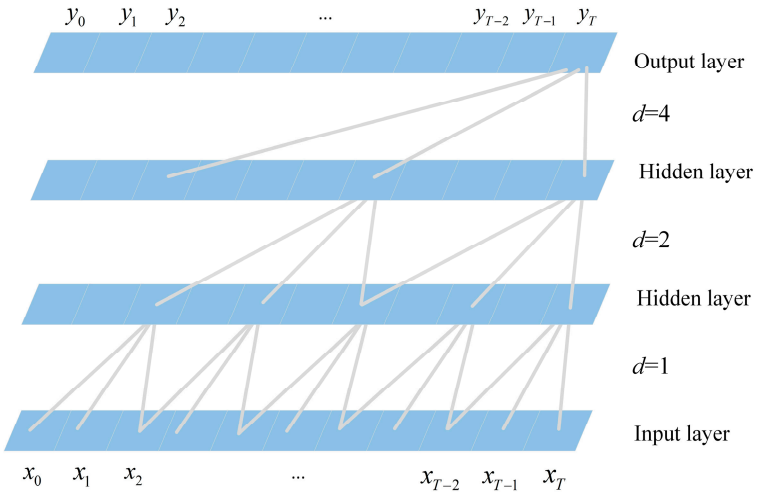


Figure 2. The dilated causal convolution structure.

When the network model becomes too deep, it is prone to the problem of degradation. The residual structure effectively alleviates this degradation issue, making the training of deep networks easier. The residual structure consists of two convolutional layers and a shortcut connection, where the output equals the sum of the outputs from the two convolutional layers plus the original input.

When handling the data characteristics of discharge voltage curves from batteries in different degradation states, a time attention mechanism is introduced into the TCN prediction model to enhance the model's focus on important time points within the time

series. The attention mechanism is a computational method that mimics human visual attention and is designed to improve the ability of a model to concentrate on important sections of the input data [36]. By assigning different attention weights to various sections of the input data, it enhances the model's performance and generalization capabilities. The time attention mechanism is an extension of the attention mechanism, suitable for time series data and sequence models. It weights the time dimension of the input sequence, allowing the model to adaptively focus on key time points and improve its ability to extract features from time series data. The process of calculating the attentional mechanism can be divided into the three following steps:

Step 1: Information input.

Step 2: Calculate the attention distribution α as

$$\alpha_t = \text{softmax}(s(x_t, q)) \tag{13}$$

where s denotes the similarity calculation between the t-th input information and the query vector q. The attention distribution vector is derived using the softmax function, and the similarity calculation used in this paper is as follows:

$$s(x_t, q) = q \cdot x_t \tag{14}$$

Step 3: Information-weighted summation, calculated as

$$att(q, X) = \sum_{t=1}^{N} \alpha_t x_t \tag{15}$$

where N represents the dimensionality of the sample, the attention distribution is calculated according to formulas (13) and (14), and Formula (15) calculates the attention value of the input message X given the query condition q.

In this paper, we propose a novel model architecture that combines a TCN with a time-attention layer, enabling the model to selectively focus on important time steps in the input sequence, thereby improving prediction performance. The architecture of this model includes several key components: Chomp1d layer, time blocks, TCN backbone, time attention layer, and fully connected output layer. To ensure that the time steps after convolution are not lost, TCN uses the padding operation, which adds additional time steps. Chomp1d trims the last few time steps in the convolution result, avoiding redundant steps at both ends of the time series output by the convolution kernel. As the basic building blocks of TCN, the time block includes two layers of dilated convolutions, followed by an activation function and Chomp1d operation. Multiple time blocks are cascaded to increase the receptive field of the convolution layer by layer, forming the TCN backbone. Finally, all layers are connected, and the final output sequence data of the model is the result of processing by all layers. The time attention layer dynamically adjusts the model's focus on different time steps by calculating attention scores, allowing the model to adapt more flexibly to variations in the sequence data. The TCN network integrated with the time attention mechanism, as illustrated in Figure 3, can more accurately reflect the dynamic changes during the voltage discharge process, thereby enhancing predictive performance.

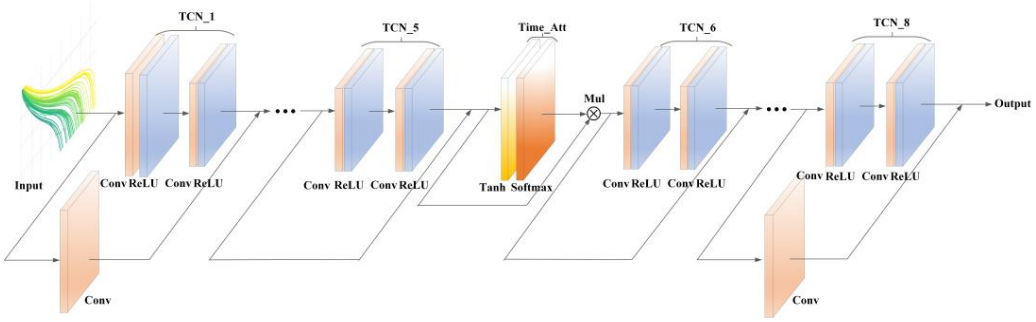


Figure 3. The network architecture diagram of ATCN model. The discharge curve of the battery serves as the input, which passes through eight layers of TCN modules, with each TCN layer employing the ReLU activation function to enhance nonlinearity. The first and last layers utilize residual connections. The features processed by the fifth TCN layer are then fed into a temporal attention layer, where attention weights are generated using the Tanh activation function and SoftMax operation. Finally, these weights are multiplied with the fifth TCN layer outputs to form the input of the next TCN layer. Among them, black dots represent the TCN layers, and blocks of different colors represent network layers with different functions.

2.3. IWOA-ATCN Prediction Model of Discharge Voltage

With the introduction of the Tent map, nonlinear convergence factors, and inertia weight in the IWOA method described in Section 2.1, this study integrates these components with the attention mechanism to construct the ATCN model. The sliding window approach is used to traverse various degraded states of battery discharge voltage curves, with the prediction target being the first value immediately following the window, thereby generating training and testing samples. The window size and sliding step are identified as hyperparameters of the ATCN model; however, manual selection of these hyperparameters is insufficient for effectively tuning the ATCN network. Therefore, the IWOA algorithm is used to optimize the ATCN parameters to develop the IWOA-ATCN prediction model.

Initially, the IWOA defines the root mean square error (RMSE) of the ATCN's overall predictions as the fitness function, which evaluates the prediction performance of the ATCN network under different configurations of sliding window size and step length. Subsequently, the IWOA algorithm is utilized to identify the hyperparameter combination that minimizes this fitness function. The flowchart of the IWOA-ATCN algorithm is illustrated in Figure 4.

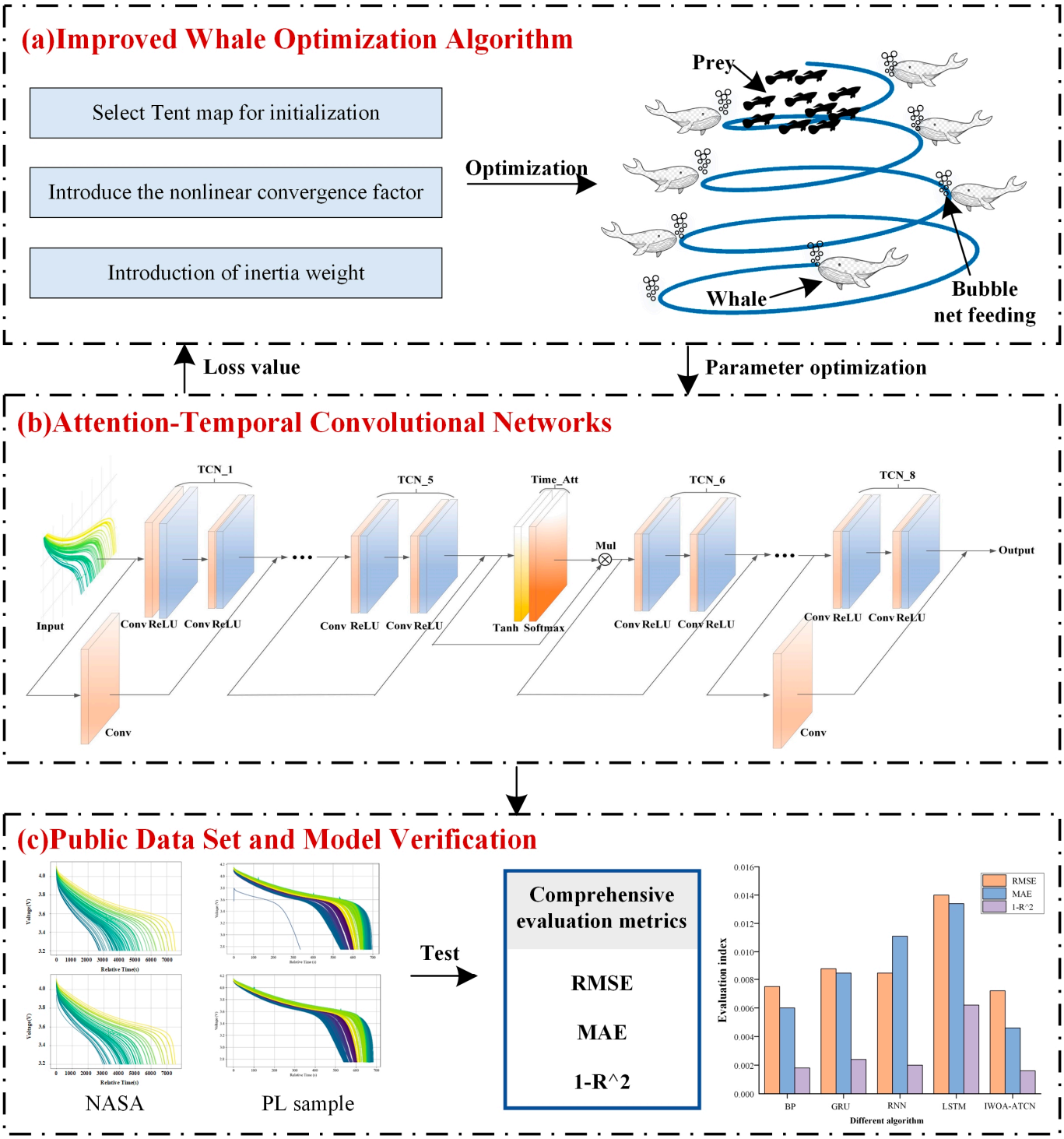


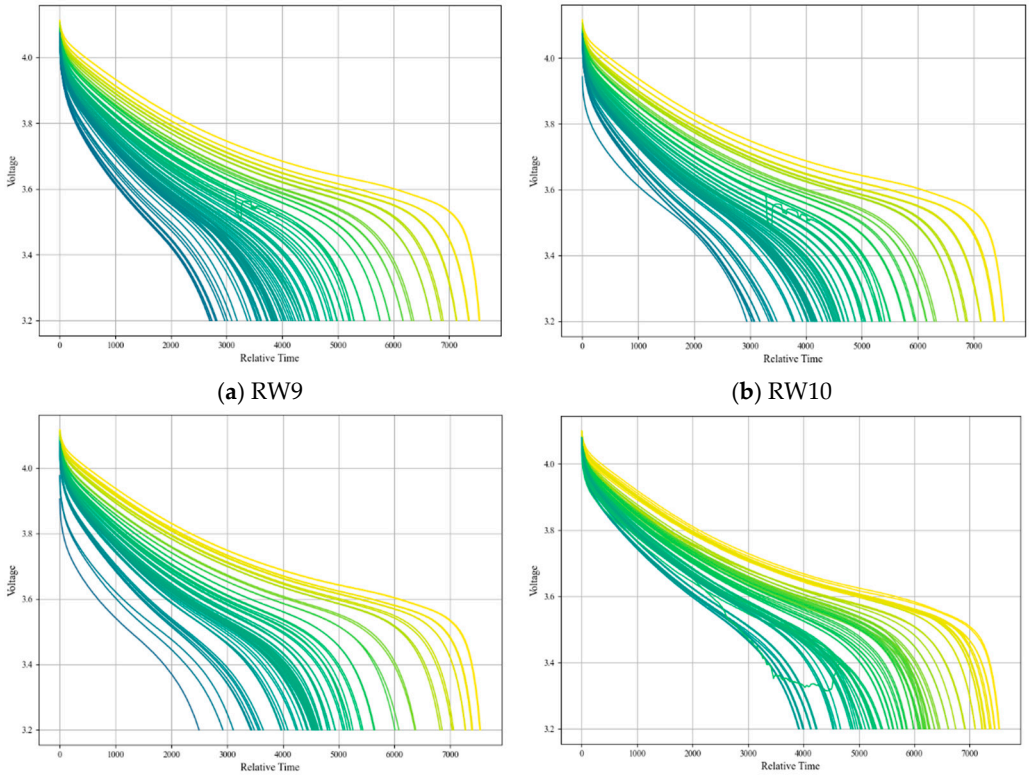
Figure 4. IWOA-ATCN model. (**a**) The Tent mapping, nonlinear convergence factor and inertia weight are introduced to improve the WOA algorithm for optimizing the window size and sliding step size. (**b**) The optimized hyperparameters and ATCN models are used for voltage prediction. Among them, black dots represent the TCN layers, and blocks of different colors represent net-work layers with different functions (**c**) The voltage prediction accuracy is validated by the publicly available dataset and evaluations. Different colors represent different degrees of degradation of the battery.

3. Experiments and Discussion

3.1. Datasets and Preparation

3.1.1. NASA

The data used in this study comes from the data repository of NASA Ames Research Center [37], which includes the voltage discharge curves of four commercial lithium-ion 18,650 batteries (RW9, RW10, RW11, and RW12). The data were collected under two load modes: random walk and reference discharge, with an experimental focus on the reference discharge mode. In the experiment, the batteries were first charged to a maximum voltage of 4.2 V; then, a current of 2 A was applied until the voltage dropped to 3.2 V. This charge–discharge process was repeated in multiple cycles to obtain the discharge voltages of the batteries in different states of degradation. As shown in Figure 5, the dataset was divided into a training set and a test set, with RW9, RW10, and RW11 as the training set, containing 234 data points, and RW12 as the test set, containing 76 data points.



(**a**) RW9 (**b**) RW10

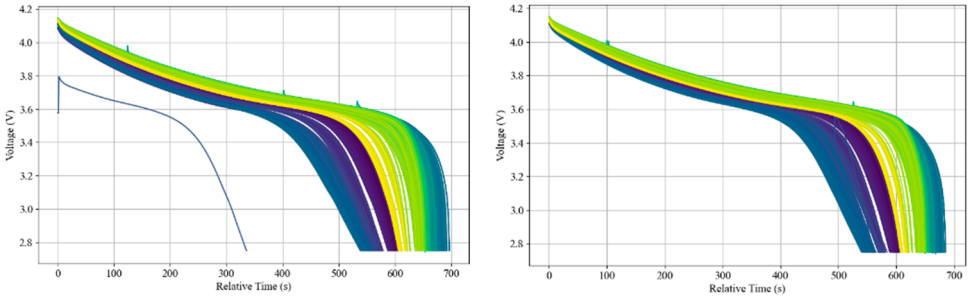
(**c**) RW11 (**d**) RW12

Figure 5. Discharge voltage curves of different batteries in the NASA dataset. Relative Time indicates that the initial time of each voltage curve is set to 0, and the end time is the relative discharge time. The change in color indicates the relative length of discharge, that is, the degree of aging state of the corresponding battery. The colour of the curve from yellow to green indicates that the relative discharge duration is from long to short, which means that the aging state of the battery is aggravated.

3.1.2. PL Sample

The PL Sample dataset encompasses partial charge–discharge cycle tests of commercial graphite/LiCoO2 pouch cells aimed at quantifying the changes in discharge voltage of batteries under different aging conditions. The tested batteries, featuring graphite/lithium cobalt oxide (LiCoO2) chemistry, have a rated capacity of 1500 mAh, and they weigh 30.3 g and measure 3.4 × 84.4 × 50.1 mm in dimensions. During the experiment, the batteries were first charged to 100% state of charge (SOC) using constant current–constant voltage (CCCV) and then discharged at a constant C/2 current until they reached the preset lower

limit of SOC (such as 0%). This paper selects data from batteries PL11 and PL13, which were tested as follows: within the 0–100% SOC range, the batteries underwent constant current charging (C/2) and constant current discharging (C/2), with a 30 min rest period set after each charge–discharge cycle to allow the batteries to recover. This process was repeated to obtain discharge voltages under different states of degradation, as shown in Figure 6. Additionally, to accurately assess the capacity fade of the batteries, full charge–discharge cycles were conducted every 50 or 100 partial cycles to update the actual discharge capacity. PL11 serves as the training set, containing 927 data points, while PL13 serves as the test set, containing 929 data points.



(**a**) PL11 (**b**) PL13

Figure 6. Discharge voltage curves of different batteries in the PL Sample dataset, in which 'Relative Time' indicates that the initial time of each voltage curve is unified to 0, and the end duration is the relative discharge time. The change in color represents the magnitude of the relative discharge time, which corresponds to the degree of aging of the respective battery. The colour of the curve goes from green to purple to blue indicating that the relative discharge duration goes from long to short, implying an aggravated state of aging of the battery.

3.2. Evaluation Metrics

In this experiment, the root mean square error (RMSE), mean absolute error (MAE), and coefficient of determination (R^2) are selected as evaluation metrics, in which RMSE is the square root of the average of the squared differences between the predicted voltage of the battery and the actual voltage, measuring the standard deviation of the model's predictive values, with the calculation formula shown in Formula (16). MAE is the average of the absolute differences between the predicted values and the actual values, reflecting the average magnitude of the error, with the calculation formula shown in Formula (17). Both RMSE and MAE are absolute indicators of prediction error size, suitable for comparing the predictive accuracy of different models. R^2; measures the relationship between the model's predicted values and the actual values, as shown in Formula (18). It is a relative indicator that reflects the model's explanatory power and goodness of fit, suitable for judging the overall performance of the model.

$$R_{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{16}$$

$$M_{MAE} = \frac{1}{N}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{17}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2} \tag{18}$$

where y_i represents the actual value of the i-th sample, $\hat{y}_i$ denotes the predicted value of the i-th sample, $\overline{y}$ is the mean of the actual values, and n is the total number of samples.

3.3. Experiment Setting

In the optimization capability test experiment of the IWOA (improved whale optimization algorithm) algorithm, a 64-bit Windows 11 system was used, with the processor type being Intel® CoreTM i5-14400F CPU @ 2.50 GHz, and the software was MATLAB R2016b. In the experiment of parameter optimization using IWOA, a GeForce RTX 4060 16GB-GPU, CUDA (v11.6), Python programming language, and Pytorch deep learning framework were utilized. The objective function was to minimize the loss rate, with parameter boundaries defined as a window size range between 10 and 100 and a sliding step size range between 1 and 20.

During the training process of TCN, the Adam optimizer was chosen, which incorporates the concepts of momentum and RMSProp by using the exponential moving average of gradients to effectively smooth the gradients and reduce the impact of noise. MSE was used as the loss function. Additionally, the experiment employed a cosine annealing decay strategy [38] to adjust the learning rate, which allows the learning rate to smoothly transition from a maximum of 0.1 to a minimum of 0.0001 during the training process, avoiding the instability in model training caused by a sudden drop in the learning rate. The cosine annealing decay strategy not only ensures that the model can learn quickly in the early stages of training but also guarantees fine-tuning in the later stages, thereby improving the model's final performance. The experiment was set with a learning cycle of 40, and a warm-up strategy was adopted in the first five cycles, which helped to make the training process more stable.

3.4. Ablation Experiment

This section presents corresponding ablation experiments for the proposed IWOA and IWOA-ATCN methods to demonstrate their effectiveness.

3.4.1. IWOA

To test the optimization capability of the IWOA proposed in this paper, six benchmark test functions are selected for testing in 10D, 50D, and 100D. Among them, f_1–f_4 are unimodal benchmark test functions and f_5–f_6 are multimodal benchmark test functions, as shown in Table 2, which includes the expressions of the functions, search ranges, and theoretical optimal values. Unimodal variable-dimensional benchmark test functions are mainly used to test the performance of the algorithm in local exploitation, whereas the multimodal variable dimensional benchmarking function is mainly employed to measure the global search capability of the algorithm. Average (Ave) and standard deviation (Std) are chosen as evaluation metrics for different algorithms, representing the mean and standard deviation of the function values separately.

In the experiment, several other optimization algorithms were selected for comparison, including Particle Swarm Optimization (PSO) [39], Grey Wolf Optimization (GWO) [40], and the standard Whale Optimization Algorithm (WOA). Additionally, several improved WOA-based algorithms proposed in this paper were included, such as WOA-A (based on the standard WOA with Tent initialization mapping), WOA-B (based on the standard WOA with nonlinear convergence), and WOA-C (based on the standard WOA with the introduction of inertia weight). The specific experimental parameter settings for the algorithms are shown in Table 3 below.

Table 2. Definition details of different types of test functions. 'U' denotes the unimodal benchmark test function type, and 'M' denotes the multimodal benchmark test function type.

Type	Name	Expression	Range	Optimal Value
U	Sphere	$f_1(x) = \sum_{i=1}^{n} x_i^2$	[−100,100]	0
	Schwefel1.2	$f_2(x) = \sum_{i=1}^{n} \left(\sum_{j=1}^{n} x_j \right)^2$	[−100,100]	0
	Step	$f_3(x) = \sum_{i=1}^{n} (\lvert x_i + 0.5 \rvert)^2$	[−100,100]	0
	Quartic	$f_4(x) = \sum_{i=1}^{n} i x_i^4 + \text{random}[0,1)$	[−128,128]	0
M	Rastrigin	$f_5(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i) + 10 \right]$	[−5.12,5.12]	0
	Griewank	$f_6(x) = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}} \right) + 1$	[−600,600]	0

Table 3. Parameter settings for each algorithm.

Algorithm	Parameter Value
WOA	A decreases linearly from 2 to 0, $p, r_1, r_2 \in [0,1]$, the spiral constant $b = 1$, and $l \in [-1,1]$
IWOA	Using the tent initialization function, a decreases nonlinearly from 2 to 0, and an inertia weight is introduced, with all other aspects the same as WOA.
WOA-A	Using the tent initialization function, with all other aspects the same as WOA.
WOA-B	A decreases nonlinearly from 2 to 0, with all other aspects the same as WOA.
WOA-C	An inertia weight is introduced, with all other aspects the same as WOA.
PSO	Learning factors $C_1 = C_2 = 1.5$ and inertia weight $\omega = 0.5$
GWO	A decreases linearly from 2 to 0, $r_1, r_2 \in [0,1]$

The population size and number of iterations for all algorithms are kept consistent with the values N = 30 and $T_{\max}$ = 500. The optimization results of the functions are shown in Table 4. As can be seen from the table, across all test functions and different dimensions, IWOA performs relatively better than other algorithms in terms of Ave and Std, especially for multimodal functions, where it can accurately obtain the theoretical optimal values in all dimensions, far outperforming PSO and GWO algorithms. This indicates that IWOA has strong global search capabilities and can avoid being trapped in local optima. In the high-dimensional cases of functions f_1 and f_3, IWOA's optimization capability is superior to WOA, and in other cases, it performs comparably to the WOA algorithm. The proposed WOA-A, WOA-B, and WOA-C show good performance in some situations, but generally, they still do not match the performance of IWOA. Furthermore, IWOA has a generally smaller Std, indicating higher consistency in results across multiple independent runs and good stability. This means that IWOA can maintain smaller error fluctuations in different runs.

The convergence curves and time consumption bar charts of each function at 50-D are shown in Figures 7 and 8, respectively. In terms of convergence speed, PSO and GWO are relatively slower compared to the other algorithms and are more prone to getting trapped in local optima in multiple runs. Regarding computational efficiency, GWO has the shorter overall time consumption, PSO and WOA take the longer, while IWOA takes the middle. In summary, although PSO and GWO perform well in some aspects, IWOA demonstrates better balance in complex optimization problems, maintaining faster convergence speed while relatively reducing the traps of local optima, providing a more reliable solution for the optimization process.

Table 4. Benchmark test function solution results, in which 'F' stands for the expressions of the functions, 'n' stands for the dimension of the functions, 'M' stands for the evaluation metrics. The bold are the ones that are closest to or equal to the theoretical values for each metric in every dimension.

F	n	M	IWOA	WOA−A	WOA−B	WOA−C	WOA	PSO	GWO
f_1	10	Ave	8.64×10^{-62}	2.51×10^{-51}	1.73×10^{-82}	4.36×10^{-44}	5.43×10^{-80}	6.84×10^{-5}	$\mathbf{2.78 \times 10^{-102}}$
		Std	3.04×10^{-61}	1.36×10^{-50}	9.49×10^{-82}	2.39×10^{-43}	2.40×10^{-79}	3.74×10^{-4}	$\mathbf{1.08 \times 10^{-101}}$
	50	Ave	5.01×10^{-57}	7.96×10^{-49}	$\mathbf{2.35 \times 10^{-80}}$	3.37×10^{-43}	1.26×10^{-79}	1.63×10^{4}	3.58×10^{-36}
		Std	2.58×10^{-56}	2.24×10^{-48}	$\mathbf{1.27 \times 10^{-79}}$	1.40×10^{-42}	4.63×10^{-79}	6.70×10^{3}	7.13×10^{-36}
	100	Ave	$\mathbf{1.32 \times 10^{-81}}$	3.19×10^{-45}	1.32×10^{-56}	5.79×10^{-44}	2.21×10^{-79}	8.06×10^{4}	4.98×10^{-24}
		Std	$\mathbf{3.91 \times 10^{-81}}$	1.49×10^{-44}	1.00×10^{-78}	2.12×10^{-43}	7.70×10^{-56}	1.30×10^{4}	1.35×10^{-23}
f_2	10	Ave	3.26×10^{-7}	9.47×10^{-8}	1.34×10^{-7}	2.23×10^{-8}	4.83×10^{-7}	$\mathbf{1.20 \times 10^{-15}}$	2.71×10^{-4}
		Std	6.91×10^{-7}	2.06×10^{-7}	2.43×10^{-7}	4.84×10^{-8}	1.81×10^{-6}	$\mathbf{6.59 \times 10^{-15}}$	5.00×10^{-4}
	50	Ave	1.25×10^{-5}	1.99×10^{-5}	1.53×10^{-5}	2.10×10^{-6}	1.19×10^{-5}	$\mathbf{3.07 \times 10^{-11}}$	5.15×10^{-3}
		Std	2.56×10^{-5}	4.67×10^{-5}	4.02×10^{-5}	5.15×10^{-6}	2.28×10^{-5}	$\mathbf{1.68 \times 10^{-10}}$	8.35×10^{-3}
	100	Ave	2.39×10^{-4}	1.39×10^{-4}	9.24×10^{-5}	1.68×10^{-5}	8.18×10^{-5}	$\mathbf{2.36 \times 10^{-7}}$	2.08×10^{-2}
		Std	4.29×10^{-4}	4.01×10^{-4}	2.78×10^{-4}	3.26×10^{-5}	2.45×10^{-4}	$\mathbf{1.29 \times 10^{-6}}$	3.37×10^{-2}
f_3	10	Ave	5.75×10^{-4}	4.76×10^{-2}	9.91×10^{-3}	6.10×10^{-2}	7.24×10^{-3}	$\mathbf{6.17 \times 10^{-11}}$	6.97×10^{-2}
		Std	5.62×10^{-4}	7.29×10^{-2}	1.83×10^{-2}	5.17×10^{-2}	1.01×10^{-2}	$\mathbf{3.30 \times 10^{-10}}$	9.99×10^{-2}
	50	Ave	$\mathbf{9.62 \times 10^{-3}}$	3.14×10^{-1}	7.29×10^{-2}	1.04	6.18×10^{-2}	1.41×10^{4}	5.82
		Std	$\mathbf{1.39 \times 10^{-2}}$	3.83×10^{-1}	1.06×10^{-1}	7.90×10^{-1}	1.19×10^{-1}	4.95×10^{3}	5.55×10^{-1}
	100	Ave	$\mathbf{2.41 \times 10^{-2}}$	7.40×10^{-1}	1.91×10^{-1}	2.02	1.44×10^{-1}	7.64×10^{4}	1.67×10
		Std	$\mathbf{4.19 \times 10^{-2}}$	7.38×10^{-1}	2.34×10^{-1}	1.65	2.11×10^{-1}	1.22×10^{4}	4.91×10^{-1}
f_4	10	Ave	8.55×10^{-4}	8.67×10^{-4}	1.20×10^{-3}	1.22×10^{-3}	7.38×10^{-4}	3.42×10^{-2}	$\mathbf{5.14 \times 10^{-4}}$
		Std	9.67×10^{-4}	1.59×10^{-3}	1.45×10^{-3}	1.64×10^{-3}	1.22×10^{-3}	2.92×10^{-2}	$\mathbf{3.88 \times 10^{-4}}$
	50	Ave	9.86×10^{-4}	1.29×10^{-3}	7.09×10^{-4}	3.31×10^{-3}	$\mathbf{5.97 \times 10^{-4}}$	9.48×10^{8}	2.96×10^{-3}
		Std	1.35×10^{-3}	2.38×10^{-3}	$\mathbf{9.09 \times 10^{-4}}$	5.65×10^{-3}	$\mathbf{9.96 \times 10^{-4}}$	5.34×10^{8}	3.14×10^{-3}
	100	Ave	$\mathbf{7.86 \times 10^{-4}}$	7.90×10^{-4}	1.24×10^{-3}	1.59×10^{-3}	1.21×10^{-3}	2.16×10^{10}	6.23×10^{-3}
		Std	9.74×10^{-4}	$\mathbf{9.04 \times 10^{-4}}$	1.84×10^{-3}	2.77×10^{-3}	1.75×10^{-3}	6.54×10^{9}	4.71×10^{-3}
f_5	10	Ave	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	1.74×10	1.72
		Std	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	8.74	4.52
	50	Ave	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	2.86×10^{2}	3.19
		Std	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	3.72×10	1.00×10
	100	Ave	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	7.99×10^{2}	2.39
		Std	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	6.23×10	7.36
f_6	10	Ave	**0.00**	3.01×10^{-2}	2.99×10^{-2}	5.32×10^{-2}	**0.00**	1.09×10^{-1}	4.57×10^{-2}
		Std	**0.00**	1.65×10^{-1}	1.15×10^{-1}	1.69×10^{-1}	**0.00**	6.69×10^{-2}	5.29×10^{-2}
	50	Ave	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	1.32×10^{2}	1.24×10^{-3}
		Std	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	5.53×10	5.28×10^{-3}
	100	Ave	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	7.18×10^{2}	3.46×10^{-3}
		Std	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	1.35×10^{2}	8.36×10^{-3}

This paper employs the IWOA for hyperparameter optimization of the ATCN, focusing primarily on the window size and the sliding step size. These two parameters are crucial for the model's performance in processing sequential data, as they directly affect the model's perception of input data and learning efficiency. For the NASA dataset and the PL Sample dataset, different numbers of whales were set to participate in the search, specifically 5, 10, and 15. Figure 9 shows the fitness values corresponding to different numbers of whales for the NASA dataset, along with the changes in the optimization parameters over the number of iterations. From Figure 9a, it is evident that an increasing number of whales results in a decreasing value of the fitness function at its end of the iteration, with the number 15 resulting in the fastest convergence and the smallest final function value. Therefore, for the NASA dataset, the experiment sets the number of whales to 15, with an optimal window size of 19 and an optimal sliding step size of 9; for the PL Sample dataset, the experiment sets the number of whales to 15, with an optimal window size of 11 and an optimal sliding step size of 4.

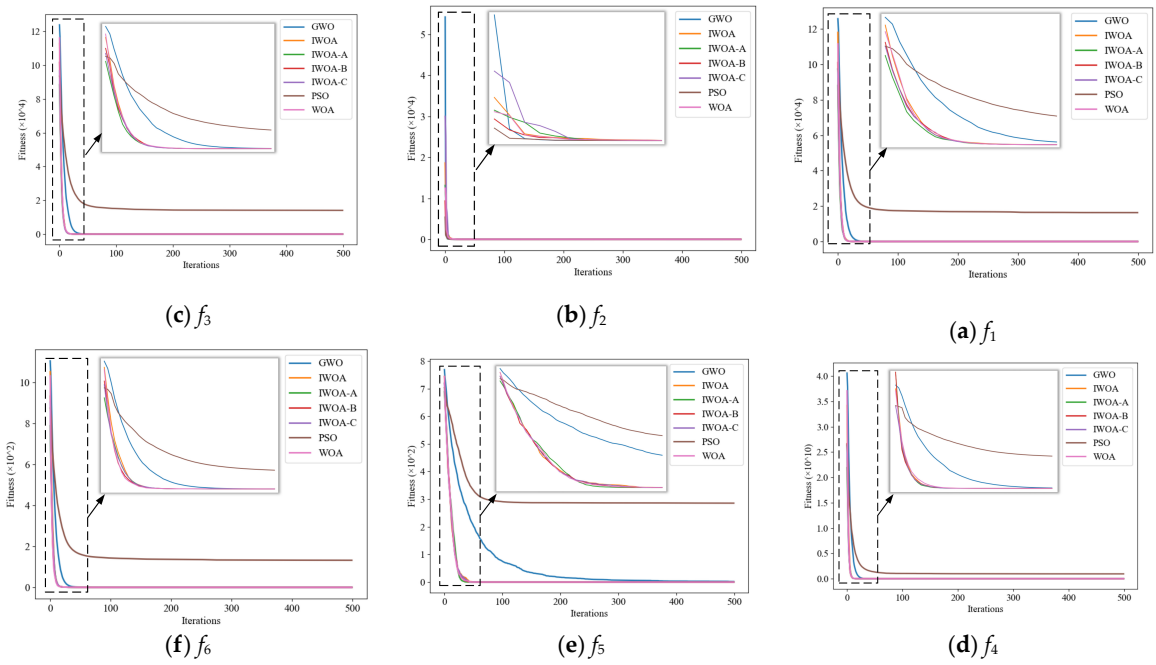


Figure 7. Fitness curves of various algorithms at 50-D.

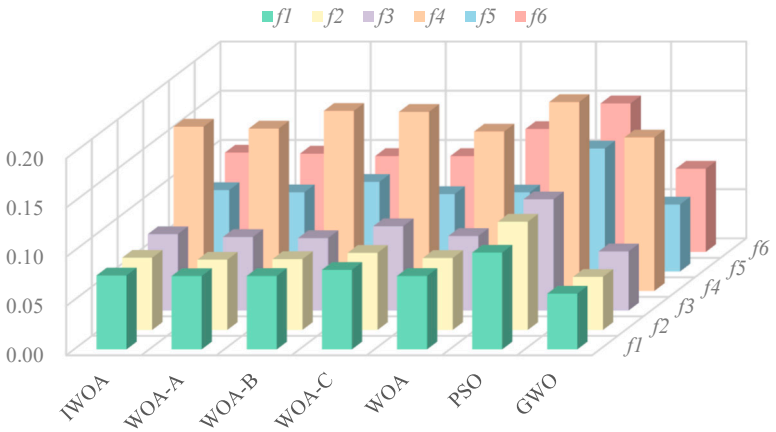
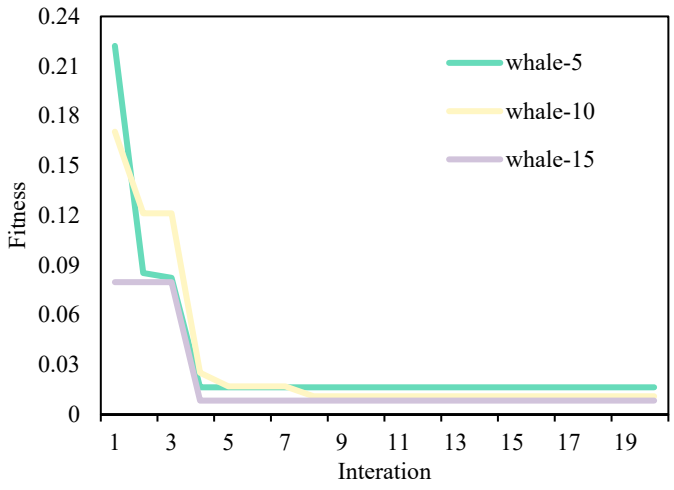


Figure 8. Bar chart of time consumption for various algorithms at 50-D. In the chart, the x-axis represents the different algorithms, the y-axis represents the 6 benchmark test functions, and the z-axis represents the duration of consumption (in seconds).



(**a**) Fitness curve variation figure.

Figure 9. *Cont.*



(**b**) Whale-5 (**c**) Whale-10 (**d**) Whale-15

Figure 9. Fitness values and their corresponding parameter changes for different numbers of whales in the NASA dataset. Among them, (**a**) shows the variation of the fitness curves with the number of iterations for different numbers of whales, while (**b**–**d**) represent the parameter changes when the number of whales is 5, 10, and 15, respectively.

3.4.2. IWOA-ATCN

(1) Ablation Experiment on Temporal Attention Layer

To demonstrate the effectiveness of the temporal attention layer and confirm its optimal placement, several ablation experiments were conducted, including configurations without temporal attention and those with temporal attention added after each layer of the TCN. The results on the NASA dataset and PL Sample dataset are presented in Tables 5 and 6, respectively. The tables indicate that adding temporal attention after any layer significantly outperforms the model without it, thereby validating the effectiveness of the proposed ATCN.

Table 5. Ablation study on temporal attention for the NASA dataset, in which 'w/o_A' indicates no temporal attention layer was added, 'xi_A' indicates that the temporal attention layer was added after the i-th layer, and 'T' represents the time consumed by the model during testing. The bold indicates the best-performing value under each metric.

Attention	Metrics			
	RMSE (%)	**MAE (%)**	**R^2 (%)**	**T (s)**
w/o_A	0.0120	0.0195	0.9849	1.12×10^{-5}
x_1_A	0.0106	0.0163	0.9871	1.62×10^{-5}
x_2_A	0.0096	0.0154	0.9975	1.59×10^{-5}
x_3_A	0.0162	0.0099	**0.9982**	1.02×10^{-5}
x_4_A	0.0079	0.0056	0.9976	$\mathbf{9.95 \times 10^{-6}}$
x_5_A	**0.0078**	**0.0054**	0.9980	1.05×10^{-5}
x_6_A	0.0132	0.0114	0.9941	9.98×10^{-6}
x_7_A	0.0113	0.0096	0.9964	2.63×10^{-5}

Furthermore, extensive experiments were carried out to identify the best placement for the temporal attention layer. The results showed that adding the temporal attention layer after the fifth layer of the TCN yields better performance with a relatively low time cost. This may be due to the fact that by the fifth layer, the model has already developed a certain level of feature extraction capability, and the introduction of the temporal attention layer can further enhance the model's focus on important temporal information, thus better adapting to the complexities of battery aging. Therefore, this study proposes to place the temporal attention layer after the fifth layer.

Table 6. Ablation study on temporal attention for the PL Sample dataset, in which 'w/o_A' indicates that no temporal attention layer was added, 'xi_A' indicates that the temporal attention layer was added after the i-th layer, and 'T' represents the duration of time the model took for testing. The bold indicates the best-performing value under each metric.

Attention	Metrics			
	RMSE (%)	MAE (%)	R^2 (%)	T(s)
w/o_A	0.0271	0.0233	0.9844	1.26×10^{-3}
x_1_A	0.0160	0.0158	0.9854	$\mathbf{9.97 \times 10^{-4}}$
x_2_A	0.0179	0.0052	0.9936	1.00×10^{-3}
x_3_A	0.0161	**0.0043**	0.9948	9.98×10^{-4}
x_4_A	0.0104	0.0056	0.9946	1.00×10^{-3}
x_5_A	**0.0090**	0.0056	**0.9955**	4.01×10^{-3}
x_6_A	0.0119	0.0061	0.9943	1.99×10^{-3}
x_7_A	0.0164	0.0089	0.9946	2.05×10^{-3}

(2) Comparison of Prediction Results

To demonstrate the effectiveness of the IWOA-ATCN prediction framework under different battery aging states, this paper trains and tests the NASA dataset and the PL Sample dataset using BPNN, GRU, RNN, and LSTM, with the parameter settings for each model as shown in Table 7.

Table 7. Model parameter settings.

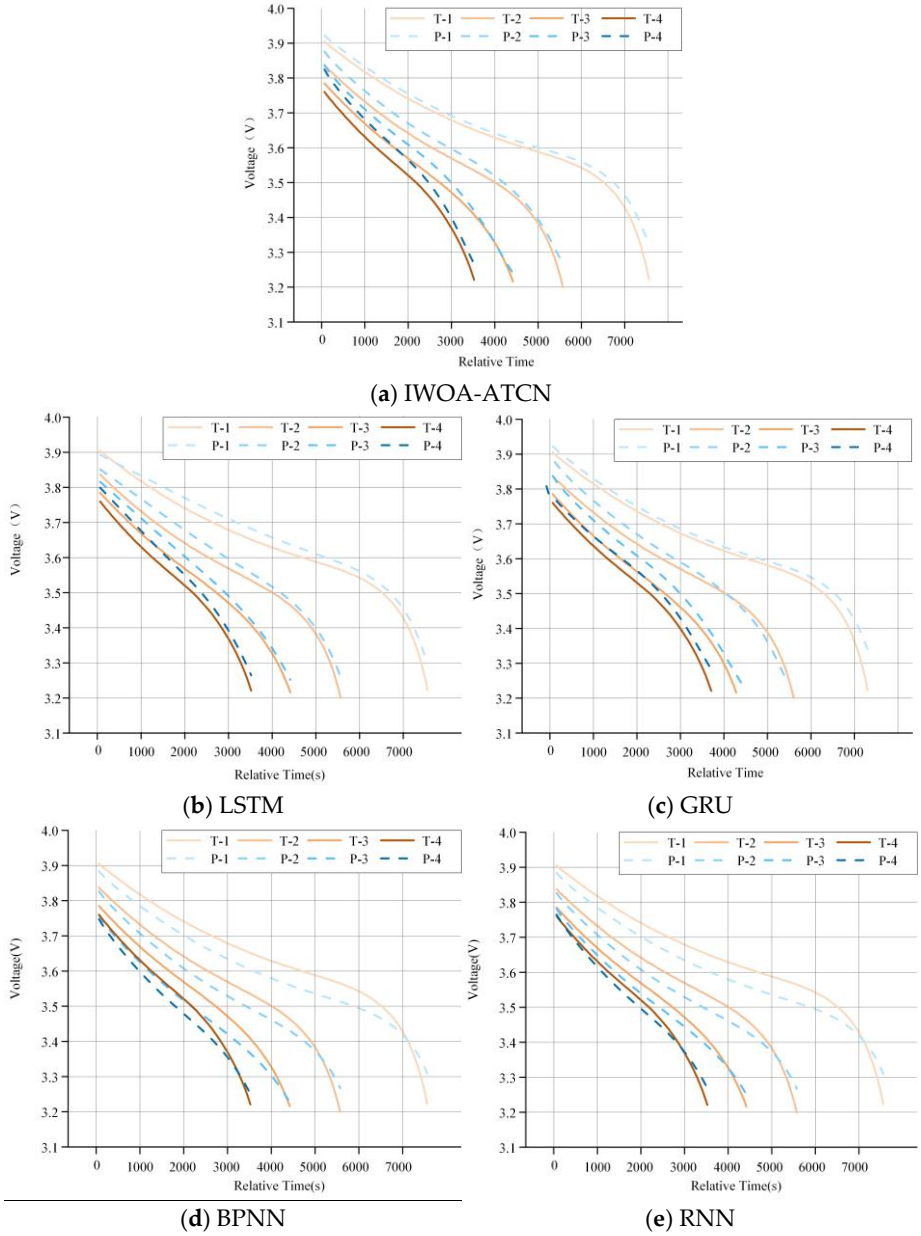
Parameter	Value or Strategy	Parameter	Value or Strategy
hidden_size	64	Activation function	RELU
learning_rate	Cosine Annealing	Loss function	MSE
epochs	40	Optimizer	Adam
batch_size	128		

(1) Results for the NASA dataset

The prediction results on the NASA dataset are presented in Figure 10, and the test results are presented in Table 8. The outcomes indicate that compared with other comparative models, the method proposed by this paper has higher prediction accuracy and lower loss rate, especially when dealing with the complex nonlinear characteristics of battery aging states, which shows a clear advantage. Additionally, the IWOA-ATCN demonstrates stronger modeling capabilities and adaptability, effectively capturing the dynamic changes in batteries at different aging stages.

Table 8. Test results of each model on the NASA dataset. The evaluation metrics include root mean square error (RMSE), Mean Absolute Error (MAE), and R-squared (R^2), presented as percentages. The bold indicates the best-performing value under each metric.

Scenarios	Evaluation Metrics		
	RMSE (%)	MAE (%)	R^2 (%)
BP	0.0075	0.0060	0.9982
GRU	0.0088	0.0085	0.9976
RNN	0.0085	0.0111	0.9980
LSTM	0.0140	0.0134	0.9938
IWOA-ATCN (this paper)	**0.0072**	**0.0046**	**0.9984**



(**a**) IWOA-ATCN

(**b**) LSTM

(**c**) GRU

(**d**) BPNN

(**e**) RNN

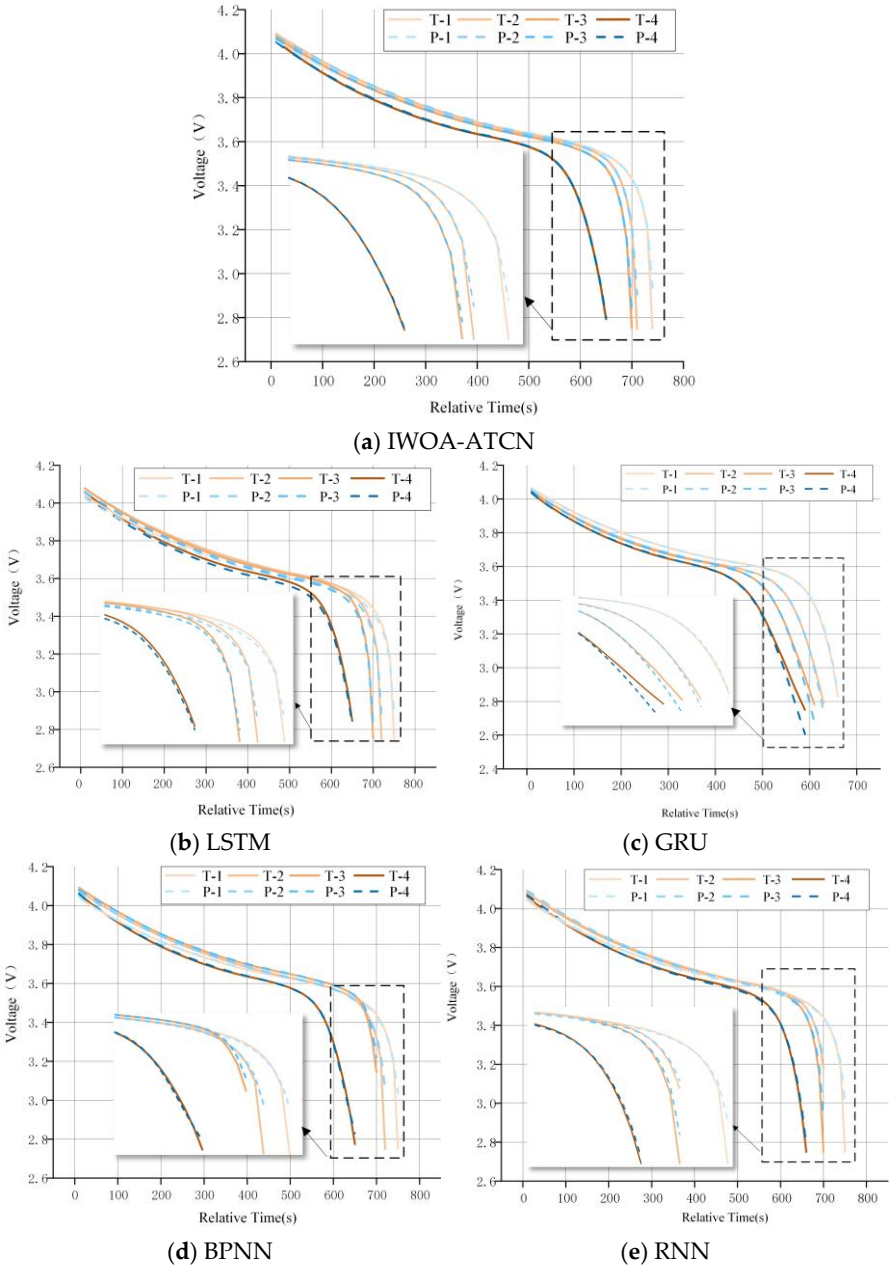
Figure 10. Visualization of the prediction results of each model on the NASA dataset. Each subfigure corresponds to a model and includes different aging states of the battery. 'T-1' represents the true value curve for aging state 1, while 'P-1' indicates the predicted value curve for the same state.

(2) Results for the PL Sample dataset

The prediction results for the PL Sample dataset are summarized in Table 9 and visualized in Figure 11, which highlights the comparative predictive performance of the proposed IWOA-ATCN method with other models. The results show that IWOA-ATCN achieved the lowest root mean square error (RMSE) of 0.0081 and the lowest mean absolute error (MAE) of 0.0040, reflecting its strong capability to explain data variance. However, its coefficient of determination (R^2) is second only to RNN, indicating that the model proposed in this paper failed to accurately capture the nonlinear characteristics of the battery discharge process. In comparison, IWOA-ATCN has more accurate predictive performance than the BPNN, GRU, RNN, and LSTM. This further emphasizes the strong modeling capability of IWOA-ATCN in capturing complex nonlinear relationships related to battery aging, demonstrating good robustness and generalization ability.

Table 9. Test results of each model on the PL Sample dataset. The evaluation metrics include root mean square error (RMSE), mean absolute error (MAE), and R-squared (R^2), presented as percentages. The bold indicates the best-performing value under each metric.

Models	Metrics		
	RMSE (%)	MAE (%)	R^2 (%)
BPNN	0.0131	0.0061	0.9966
GRU	0.0185	0.0177	0.9932
RNN	0.0093	0.0076	**0.9983**
LSTM	0.0159	0.0045	0.9950
IWOA-ATCN (this paper)	**0.0081**	**0.0040**	0.9974



(**a**) IWOA-ATCN

(**b**) LSTM

(**c**) GRU

(**d**) BPNN

(**e**) RNN

Figure 11. Visualization of the prediction results of each model on the PL Sample dataset. Each subfigure corresponds to a model and includes different aging states of the battery. 'T-1' represents the true value curve for aging state 1, while 'P-1' indicates the predicted value curve for the same state.

3.5. Discussion

In the experimental section of this study, the performance of the IWOA-ATCN prediction framework was systematically evaluated under different battery aging states.

In terms of parameter optimization, IWOA demonstrated strong global search capabilities, accurately obtaining theoretical optimal values on multi-modal functions, significantly outperforming PSO and GWO. It also exhibited excellent stability and moderate convergence speed. By optimizing the window size and sliding step of the TCN model, IWOA successfully identified the best parameter configurations suitable for the NASA and PL Sample datasets, significantly enhancing the model's learning ability and prediction accuracy.

Regarding voltage prediction performance, the IWOA-ATCN model outperformed other models in terms of prediction accuracy and loss rate on the NASA dataset, particularly showcasing a strong ability to capture dynamic changes when dealing with complex battery aging states. On the PL Sample dataset, although the model's capability to capture nonlinear features was slightly weaker, IWOA-ATCN still demonstrated outstanding robustness and generalization ability, achieving the lowest RMSE and MAE metrics.

Additionally, ablation experiments validated the effectiveness of the temporal attention layer; regardless of its configuration within the TCN, the model's performance improved significantly. The experimental results indicated that placing the temporal attention layer after the fifth layer of the TCN yielded the best results, enhancing the model's focus on important temporal information and better adapting to the complexities of battery aging.

4. Conclusions

This paper proposes a battery discharge voltage prediction model based on IWOA-ATCN to improve prediction accuracy under different battery degradation states. The IWOA method demonstrates significant advantages in parameter optimization and voltage prediction performance. By optimizing the window size and sliding step of the TCN model, IWOA successfully identified the best parameter configurations suitable for the NASA dataset, with a window size of 19 and a sliding step of 9, and for the PL Sample dataset, with a window size of 11 and a sliding step of 4. Additionally, ablation experiments validated the effectiveness of the temporal attention layer, enhancing the model's ability to focus on important temporal information, which better adapts to the complexities of battery aging and significantly improves the model's learning capability and voltage prediction accuracy under complex battery aging states. The model was tested on the NASA dataset with an RMSE of 0.0072, MAE of 0.0046, and R^2 of 0.9984. The test results on the PL Sample dataset showed RMSE of 0.0081, MAE of 0.0040, and R^2 of 0.9983. It is indicated that the prediction performance of the IWOA-ATCN model is higher than other models such as BP, RNN, and LSTM.

With the continuous development of smart grid and electric vehicle technologies, the IWOA-ATCN model is expected to play an important role in practical applications, promoting the safe and efficient operation of energy storage systems. Therefore, further research should focus on enhancing the model's generalization ability and practicality to adapt to the ever-changing market demands and technological challenges. Future studies could incorporate more types of data sources, such as temperature and load, to improve the model's comprehensive understanding and predictive capability regarding battery states, as well as explore more efficient optimization algorithms to reduce resource consumption.

Author Contributions: Conceptualization: J.Y.; methodology: J.Y.; validation, J.Y. and G.W.; formal analysis, Q.H. and Y.C.; investigation, Z.Y.; resources, Y.H.; data curation, G.W.; writing—original draft preparation, L.L.; writing—review and editing, J.Y.; visualization, Q.H.; supervision, Y.C.; project administration, Z.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by "The Key Technology Projects of China Southern Power Grid", grant number ZDKJXM20222029". The funder is China Southern Power Grid Company Limited.

Data Availability Statement: The data used in this study were obtained from the NASA Ames Research Center's data repository and PL Sample of the CALCE battery team.

Conflicts of Interest: All Authors were employed by the company Power Dispatching & Control Center of China Southern Power Grid. The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abbreviations

IWOA	Improved Whale Optimization Algorithm
ATCN	Attention Temporal Convolutional Network
BP	Back Propagation Neural Network
RNN	Recurrent Neural Network
GRU	Gated Recurrent Unit
LSTM	Long Short-Term Memory
BMS	Battery Management System
SOC	State of Charge
FNN	Feedforward Neural Network
CNN	Convolutional Neural Network
$X_{\text{best}}(t)$	The optimal whale position at the t-th iteration
D	The distance from the prey to the whale
$X(t+1)$	The position of the whale at the $(t+1)$-th iteration
t	The number of iterations
a	The convergence factor
$T_{\max}$	The maximum number of iterations
$X_{rand}(t)$	The current randomly selected whale position
m	The number of the whale
x_{m+1}	The position of the $(m+1)$-th whale
n	The factor that controls the nonlinearity
$\omega(t)$	The inertia weights
α	The attention distribution
q	The query vector
N	The dimensionality of the sample
RMSE	The root mean square error
MAE	Mean absolute error
R^2	Coefficient of determination
y_i	The actual value of the i-th sample
$\hat{y}_i$	The predicted value of the i-th sample
$\overline{y}$	The mean of the actual values
PSO	Particle Swarm Optimization
GWO	Grey Wolf Optimization
WOA-A	The standard WOA with Tent initialization mapping
WOA-B	The standard WOA with nonlinear convergence
WOA-C	The standard WOA with the introduction of inertia weight
U	The unimodal benchmark test function type
M	The multimodal benchmark test function type
w/o_A	Without temporal attention layer
xi_A	The temporal attention layer was added after the i-th layer

References

1. Li, K.; Wei, F.; Tseng, K.J.; Soong, B.-H. A Practical Lithium-Ion Battery Model for State of Energy and Voltage Responses Prediction Incorporating Temperature and Ageing Effects. *IEEE Trans. Ind. Electron.* **2018**, *65*, 6696–6708. [CrossRef]
2. Chen, W.; Liang, J.; Yang, Z.; Li, G. A Review of Lithium-Ion Battery for Electric Vehicle Applications and Beyond. *Energy Procedia* **2019**, *158*, 4363–4368. [CrossRef]
3. Xiong, R.; Sun, Y.; Wang, C.; Tian, J.; Chen, X.; Li, H.; Zhang, Q. A data-driven method for extracting aging features to accurately predict the battery health. *Energy Storage Mater.* **2023**, *57*, 460–470. [CrossRef]
4. Han, X.; Lu, L.; Zheng, Y.; Feng, X.; Li, Z.; Li, J.; Ouyang, M. A review on the key issues of the lithium ion battery degradation among the whole life cycle. *eTransportation* **2019**, *1*, 100005. [CrossRef]
5. Winata, H.; Surantha, N. Online voltage and degradation value prediction of lead acid battery using gaussian process regression. *Appl. Sci.* **2023**, *13*, 12059. [CrossRef]
6. Rao, Z.; Wu, J.; Li, G.; Wang, H. Voltage abnormity prediction method of lithium-ion energy storage power station using informer based on bayesian optimization. *Sci. Rep.* **2024**, *14*, 21404. [CrossRef] [PubMed]
7. Zhao, M.; Zhang, Y.; Fang, S. Improving in-situ life prediction and classification performance by capturing both the present state and evolution rate of battery aging. *J. Energy Storage* **2024**, *83*, 110768. [CrossRef]
8. Song, Y.; Liu, D.; Liao, H. A hybrid statistical data-driven method for on-line joint state estimation of lithium-ion batteries. *Appl. Energy* **2020**, *261*, 114408. [CrossRef]
9. Kawahara, T.; Sato, K.; Sato, Y. Battery voltage prediction technology using machine learning model with high extrapolation accuracy. *Int. J. Energy Res.* **2023**, *2023*, 5513446. [CrossRef]
10. Hu, X.; Xu, L.; Lin, X. Battery lifetime prognostics. *Joule* **2020**, *4*, 310–346. [CrossRef]
11. Lipu, M.S.H.; Hannan, M.A.; Hussain, A. A review of state of health and remaining useful life estimation methods for lithium-ion battery in electric vehicles: Challenges and recommendations. *J. Clean. Prod.* **2018**, *205*, 115–133. [CrossRef]
12. Shu, X.; Shen, S.; Shen, J. State of health prediction of lithium-ion batteries based on machine learning: Advances and perspectives. *iScience* **2021**, *24*, 103265. [CrossRef] [PubMed]
13. Xue, Q.; Li, J.; Xu, P. Machine learning based swift online capacity prediction of lithium-ion battery through whole cycle life. *Energy* **2022**, *261*, 125210. [CrossRef]
14. Barré, A.; Deguilhem, B.; Grolleau, S. A review on lithium-ion battery aging mechanisms and estimations for automotive applications. *J. Power Sources* **2013**, *241*, 680–689. [CrossRef]
15. Zhang, Y.; Liu, J.; Wang, Q. Battery State-of-Charge Estimation Based on a Modified Whale Optimization Algorithm. *IEEE Trans. Ind. Electron.* **2022**, *69*, 4100–4109.
16. Robert, R.R.; Michael, A.O.; David, A.H. Battery health prediction under generalized conditions using a Gaussian process transition model. *J. Energy Storage* **2019**, *23*, 320–328.
17. Lin, P.; Jin, P.; Hong, J.; Wang, Z. Battery voltage and state of power prediction based on an improved novel polarization voltage model. *Energy Rep.* **2020**, *6*, 2299–2308. [CrossRef]
18. Monteiro, P.G.A. Data-Driven State Estimation Algorithms for A 48V Battery Management System. Ph.D. Thesis, Universidade do Porto (Portugal), Porto, Portugal, 2021.
19. Xiang, Y.; Fan, W.; Zhu, J.; Wei, X.; Dai, H. Semi-supervised deep learning for lithium-ion battery state-of-health estimation using dynamic discharge profiles. *Cell Rep. Phys. Sci.* **2024**, *5*, 101763. [CrossRef]
20. Fan, Y.; Xiao, F.; Li, C.; Yang, G.; Tang, X. A novel deep learning framework for state of health estimation of lithium-ion battery. *J. Energy Storage* **2020**, *32*, 101741. [CrossRef]
21. Luca, B.; Tommaso, B.; Chetan, K.; Olga, F. Ageing-aware battery discharge prediction with deep learning. *Appl. Energy* **2023**, *346*, 121229.
22. Tran, M.K.; Panchal, S.; Chauhan, V. Python-based scikit-learn machine learning models for thermal and electrical performance prediction of high-capacity lithium-ion battery. *Int. J. Energy Res.* **2022**, *46*, 786–794. [CrossRef]
23. Hong, J.; Wang, Z.; Yao, Y. Fault prognosis of battery system based on accurate voltage abnormity prognosis using long short-term memory neural networks. *Appl. Energy* **2019**, *251*, 113381. [CrossRef]
24. Zhao, R.; Kollmeyer, P.J.; Lorenz, R.D. A compact unified methodology via a recurrent neural network for accurate modeling of lithium-ion battery voltage and state-of-charge. In Proceedings of the 2017 IEEE Energy Conversion Congress and Exposition (ECCE), Cincinnati, OH, USA, 7 November 2017.
25. Heinrich, F.; Klapper, P.; Pruckner, M. A comprehensive study on battery electric modeling approaches based on machine learning. *Energy Inform.* **2021**, *4*, 17. [CrossRef]
26. Lu, J.; Xiong, R.; Tian, J. Battery degradation prediction against uncertain future conditions with recurrent neural network enabled deep learning. *Energy Storage Mater.* **2022**, *50*, 139–151. [CrossRef]
27. Lu, J.; Xiong, R.; Tian, J. Deep learning to predict battery voltage behavior after uncertain cycling-induced degradation. *J. Power Sources* **2023**, *581*, 233473. [CrossRef]

28. Bai, S.; Kolter, J.Z.; Koltun, V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv* **2018**, arXiv:1803.01271.
29. Lea, C.; Vidal, R.; Reiter, A. Temporal Convolutional Networks: A Unified Approach to Action Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Amsterdam, The Netherlands, 8–10 and 15–16 October 2016.
30. Xu, X.; Mao, H.; Zhao, Y. An urban traffic flow fusion network based on a causal spatiotemporal graph convolution network. *Appl. Sci.* **2022**, *12*, 7010. [CrossRef]
31. Zhan, X.; Li, Y.; Li, R.; Gu, X.; Habimana, O.; Wang, H. Stock price prediction using time convolution long short-term memory network. In Proceedings of the Knowledge Science, Engineering and Management, KSEM, Rome, Italy, 15–18 October 2018.
32. Zhang, W.; Li, T.; Li, P. Application of spatiotemporal convolution network for precipitation nowcasting. In Proceedings of the Thirteenth International Conference on Graphics and Image Processing (ICGIP 2021), Zhangjiajie, China, 19–21 November 2021.
33. Lin, Y.; Koprinska, I.; Rana, M. Temporal Convolutional Attention Neural Networks for Time Series Forecasting. In Proceedings of the International Joint Conference on Neural Networks (IJCNN), Shenzhen, China, 18–22 July 2021.
34. Bowei, L.; Mengzui, D. Enterprise Financial Early Warning Based on Improved Whale Optimization Algorithm: Optimize the Perspective with Indicators. *Sci. Program.* **2022**, *2022*, 1457547.
35. Mafarja, M.; Jaber, I.; Ahmed, S. Whale Optimisation Algorithm for high-dimensional small-instance feature selection. *Int. J. Parallel Emergent Distrib. Syst.* **2021**, *36*, 80–96. [CrossRef]
36. Yang, K.; Tang, Y.; Zhang, S. A deep learning approach to state of charge estimation of lithium-ion batteries based on dual-stage attention mechanism. *Energy* **2022**, *244*, 123233. [CrossRef]
37. Bole, B.; Kulkarni, C.; Daigle, M. *Randomized Battery Usage Data Set. NASA Ames Prognostics Data Reposito*; NASA: Mountain view, CA, USA, 2014.
38. Loshchilov, I.; Hutter, F. SGDR: Stochastic Gradient Descent with Warm Restarts. *arXiv* **2016**, *1608*, 03983v5.
39. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; pp. 1942–1948.
40. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]

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