

Frontiers of Information Technology & Electronic Engineering
 www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
 ISSN 2095-9184 (print); ISSN 2095-9230 (online)
 E-mail: jzus@zju.edu.cn



Dynamic time prediction for electric vehicle charging based on charging pattern recognition

Chunxi LI¹, Yingying FU¹, Xiangke CUI², Quanbo GE^{†3,4,5}

¹Logistics Engineering College, Shanghai Maritime University, Shanghai 200135, China

²School of Economics and Management, Beijing Jiaotong University, Beijing 102603, China

³School of Automation, Nanjing University of Information Science and Technology, Nanjing 210044, China

⁴Jiangsu Collaborative Innovation Center of Atmospheric Environment and Equipment Technology (CICAEEET), Nanjing University of Information Science and Technology, Nanjing 210044, China

⁵Jiangsu Key Laboratory of Big Data Analysis Technology (B-DAT), Nanjing University of Information Science and Technology, Nanjing 210044, China

E-mail: lcx46@163.com; Fuyy8652@163.com; 18458320@qq.com; QuanboGe@163.com

Received May 16, 2022; Revision accepted Aug. 23, 2022; Crosschecked Dec. 12, 2022

Abstract: Overcharging is an important safety issue in the charging process of electric vehicle power batteries, and can easily lead to accelerated battery aging and serious safety accidents. It is necessary to accurately predict the vehicle's charging time to effectively prevent the battery from overcharging. Due to the complex structure of the battery pack and various charging modes, the traditional charging time prediction method often encounters modeling difficulties and low accuracy. In response to the above problems, data drivers and machine learning theories are applied. On the basis of fully considering the different electric vehicle battery management system (BMS) charging modes, a charging time prediction method with charging mode recognition is proposed. First, an intelligent algorithm based on dynamic weighted density peak clustering (DWDPC) and random forest fusion is proposed to classify vehicle charging modes. Then, on the basis of an improved simplified particle swarm optimization (ISPSO) algorithm, a high-performance charging time prediction method is constructed by fully integrating long short-term memory (LSTM) and a strong tracking filter. Finally, the data run by the actual engineering system are verified for the proposed charging time prediction algorithm. Experimental results show that the new method can effectively distinguish the charging modes of different vehicles, identify the charging characteristics of different electric vehicles, and achieve high prediction accuracy.

Key words: Charging mode; Charging time; Random forest; Long short-term memory (LSTM); Simplified particle swarm optimization (SPSO)

<https://doi.org/10.1631/FITEE.2200212>

CLC number: U469.72; TP391.4

1 Introduction

With steady growth in the number of electric vehicles (EVs) in the automobile market, the running problems and charging safety of EVs are important issues to be solved at present. Among many battery

problems, overcharging of EVs is one of the most frequently encountered safety problems (Zhu XQ et al., 2019). Due to aging, inconsistency, battery management system (BMS) failures, and other reasons, EV batteries are prone to overcharging, which leads to the loss of anode and cathode materials and excessive electrolyte decomposition. This deterioration leads to thermal runaway of the battery, which causes an explosion and fire (Li CX et al., 2020). The

[†] Corresponding author

ORCID: Chunxi LI, <https://orcid.org/0000-0003-1055-4755>;
 Quanbo GE, <https://orcid.org/0000-0002-6907-7837>

© Zhejiang University Press 2023

charging pile is the main charging device for EVs, and can directly affect the EV charging behavior. The accurate prediction of charging time by the charging pile can effectively prevent battery from overcharging. Therefore, the prediction of EV charging time is of great significance in the operation and charging safety of EVs. Data used in this study are obtained from charging piles, and the EV charging time is also predicted from the perspective of charging piles to prevent overcharging.

The traditional method of predicting remaining charging time calculates the required charging time per unit state of charge (SOC) according to the total charging time, and calculates the remaining charging time according to the required charging SOC (Sun et al., 2021). This method is not suitable for studying the charging time of EVs because of the uneven distribution of EV charging time. An equivalent circuit model can be established to study the charging condition of the battery and reflect its charging behavior (Frendo et al., 2020). However, the establishment of an equivalent circuit model requires detailed knowledge of battery parameters and battery pack structure, which is difficult to obtain by charging pile, and it is challenging to determine battery parameters through experiments, which requires considerable engineering work (Frendo et al., 2020). Moreover, due to battery inconsistency, when an electric car has not reached the full charge state, the phenomenon of overcharging may have occurred. EV BMSs charge according to the unique EV charging mode, and the charging voltage and current are different in different charging modes, which leads to great differences in charging time. All these factors increase the difficulty in predicting EV charging time (Zhang QS and Zhao, 2020).

With the development of artificial intelligence, machine learning algorithms have been applied to various prediction methods (Althé and de la Fortelle, 2017; Roondiwala et al., 2017; Chang et al., 2020; Liu WW et al., 2022; Yang et al., 2022). Compared with the physical battery model, the machine learning method has a lower computational cost. The data-driven machine learning model represents the latest development in prediction methods. In Liu YY (2018), a battery model with adaptive online parameters based on Lyapunov's law was used to evaluate the SOC of the lithium battery, but the relationship between it and the charging time was

not explained. The charging time of the battery can be predicted according to the charging temperature (Zhu ZC and Zheng, 2017). The empirical model established by this method can predict the battery charging time at different temperatures, but the model is strongly dependent on data features and thus has weak generalization ability. In Cheng et al. (2019), fuzzy information granulation (IG) was combined with a support vector regression (SVR) model to predict the remaining charging time of the lithium battery. This method improves the training efficiency, but it has limited ability to represent complex functions when there are limited samples. The remaining time can be predicted using the unit charging power method (Han et al., 2014); however, this method requires a large amount of collection and calculation work, and thus is not suitable for practical engineering applications. In Lin (2018), a three-stage SVR model was used to predict the remaining charging time of the lithium battery; it divides the charging process into an early stage, middle stage, and late stage. In Zhou et al. (2019), the estimation of battery capacity was converted into the estimation of the time required for full charging of the battery. The real-time full charging time of the battery can be predicted using an extended Kalman filter and Gaussian process regression (EKF-GPR) algorithm. Still, the error will gradually increase with the increase of the number of cycles. Independent long short-term memory (IndyLSTM) was used to predict the remaining charging time of the lithium battery in Liu X (2020), but the charging methods of various batteries were not considered, and the parameters on which the algorithm depends needed to be selected artificially.

At present, most research studies only the charging time of a single battery, and does not consider the impact of vehicle charging mode, so it cannot be applied well to the EV industry. It is necessary to propose an intelligent charging time prediction algorithm that can identify different EV charging patterns. To fill this research gap, in this paper we add a charging mode recognition module to the prediction algorithm, and study the constant current constant voltage (CC-CV) mode and multistage constant current (MCC) mode. These are the main modes used by EVs in the market, making the research content more universal. Different from other work, we use real charging vehicle data, and thus the situation is

more realistic and the model is more suitable for the general vehicle type.

2 Problem description

2.1 System description

As is commonly known, predicting the charging time of EVs is a time series prediction problem. LSTM performs well in time series data and is often used as a charging time prediction method. An LSTM schematic and the calculation equations are shown in Fig. 1 and Eqs. (1)–(6) (Zhang YF et al., 2021):

$$f_t = \sigma(W_f h_{t-1} + W_f x_t + b_f), \quad (1)$$

$$i_t = \sigma(W_i h_{t-1} + W_i x_t + b_i), \quad (2)$$

$$a_t = \tanh(W_c h_{t-1} + W_c x_t + b_c), \quad (3)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot a_t, \quad (4)$$

$$o_t = \sigma(W_o h_{t-1} + W_o x_t + b_o), \quad (5)$$

$$h_t = o_t \odot \tanh(c_t). \quad (6)$$

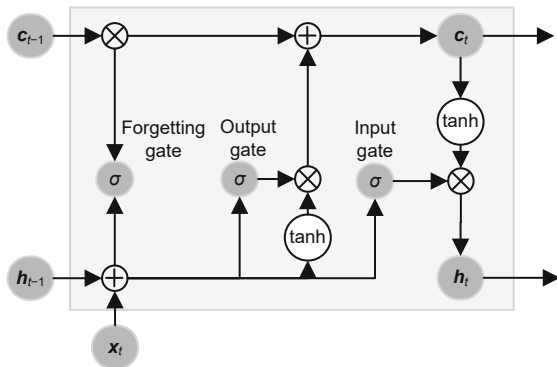


Fig. 1 Unit structure of long short-term memory (LSTM)

Eqs. (1)–(6) are the equations for predicting the charging time of traditional LSTM, where x is the charging input data, h is the output forecasting data, “ \odot ” represents the inner product of vectors, c is the state of the unit, and t is the time. f , i , and o are the forgetting gate, input gate, and output gate, respectively. σ denotes the sigmoid activation function, and \tanh denotes the tangent function. W and b represent the weight matrix and the deviation vector, respectively (the subscripts “f,” “i,” “c,” and “o”

refer to the forgetting gate, input gate, state of the unit, and output gate, respectively).

In the actual charging scenario, due to the complexity of vehicle models and the great differences in charging methods, for a large number of different vehicles, using a single LSTM method cannot obtain better prediction accuracy or solve the problem of preventing overcharge. In addition, LSTM model parameters must be determined, such as the number of neurons, the learning rate, and the number of iterations. The learning rate controls the learning speed of LSTM. If the learning rate is too low, the convergence time of the algorithm will be too long. If the learning rate is too high, the learning speed of the algorithm will be too high, and the model may miss the optimal value, which makes it difficult to achieve sufficient accuracy. Similarly, too few or too many neurons and iterations will affect the convergence speed and prediction accuracy of the algorithm. These problems make it difficult to determine the parameters of the prediction algorithm. In Liu B (2020), the parameters of the LSTM prediction model were determined by particle swarm optimization (PSO). PSO is a simple optimization algorithm, and has been used for parameter optimization of various algorithms because of its being simple and efficient. The LSTM method improved by PSO is shown in Fig. 2.

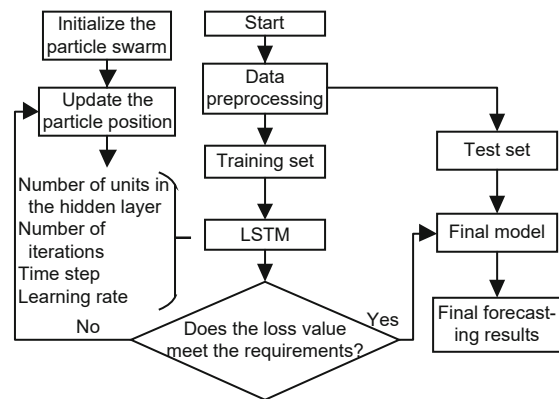


Fig. 2 Flowchart of PSO-LSTM (PSO: particle swarm optimization; LSTM: long short-term memory)

Generally, the particle l in the particle swarm represents the parameters of LSTM, including the number of units in the hidden layer, number of iterations, time step, and learning rate. Hu and Li (2007) proved that the PSO evolution process is independent of particle velocity, and that the ultimate

goal of the algorithm is to make the current solution infinitely close to the optimal solution. Therefore, they proposed a simplified PSO (SPSO) algorithm, whose calculation equation is

$$\mathbf{l}_{id}^{k+1} = \omega \mathbf{l}_{id}^k + c_1 r_1 (\mathbf{p}_{id} - \mathbf{l}_{id}^k) + c_2 r_2 (\mathbf{g}_{bd} - \mathbf{l}_{id}^k). \quad (7)$$

In Eq. (7), \mathbf{l}_{id} is the position of the i^{th} particle in dimension, \mathbf{p}_{id} is the current local optimal position of the particle, and \mathbf{g}_{bd} is the global optimal position of the particle swarm. c_1 and c_2 are learning factors, and r_1 and r_2 are random numbers between 0 and 1. ω is a non-negative inertia weight and k represents the moment.

However, in the face of solving complex multi-peak and multi-local problems, the particle optimization ability of SPSO will be reduced due to the lack of speed constraints, which will affect the rate of convergence and the accuracy of the algorithm.

2.2 Research motivation

Different EVs have different BMSs and charging modes. At present, other research has not considered the identification of charging mode when studying the charging behavior of EV batteries. The EV charging mode represents the change of charging current with time. The CC-CV and MCC modes are the most common charging modes, as shown in Fig. 3.

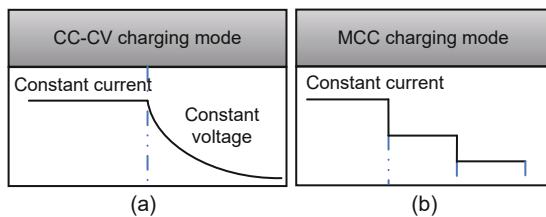


Fig. 3 Mainstream charging modes of an electric vehicle (EV): (a) constant current constant voltage (CC-CV) charging mode; (b) multistage constant current (MCC) charging mode

To improve the accuracy of charging time prediction, the charging mode should be considered and a clustering method can be used to classify modes. Density peak clustering (DPC) is a density-based clustering method proposed by Rodriguez and Laio (2014); it can classify similar data into the same cluster and realize efficient clustering among arbitrary data. However, DPC is not sensitive to the local set of data and performs poorly, especially when the internal data change greatly. Although k nearest

neighbors (KNN) can be used to improve DPC (Du et al., 2016), its Euclidean distance cannot reflect the importance of each indicator. In this study we propose a new KNN-DWDPC (dynamic weight density peak clustering) method, which can solve the problem of local insensitivity of DPC to data, and increase the attribute differentiation among data.

LSTM is selected to predict EV charging time in this study. To solve the problem that the LSTM accuracy is affected by manual parameter setting, it is necessary to select an intelligent optimization algorithm to determine the appropriate parameters. We propose a new improved simplified particle swarm optimization (ISPSO) algorithm based on SPSO and the velocity mean. This method not only enlarges the scope of particle search, but also makes it easier for the particle to approach the global optimal value. To eliminate the problem that ISPSO falls into a local optimum due to the lack of a velocity term, a strong tracking filter is used to improve the updating position of the particle swarm. Compared with existing prediction algorithms, the improved fusion algorithm can effectively avoid the shortcoming of the PSO algorithm, which cannot provide suitable parameters for LSTM due to local optimization. Finally, the improved fusion algorithm is applied to the charging vehicles in different charging modes to predict charging time in real time.

The proposed EV charging time prediction method (Fig. 4) is divided mainly into three modules, i.e., the data collection module, classification module, and prediction module. The main contributions of this study are as follows:

1. To reduce the large classification error caused by the significant variation in charging current in the same charging mode, an improved DWDPC method is proposed based on KNN. This method can improve the poor performance of DPC caused by different local data densities, and can make the classification samples more representative by dividing the charging data into several similar data sets.

2. Focusing on the multi-mode charging characteristics of EVs and the fact that prediction of charging time is not accurate in a single mode, a charging pile charging mode feature recognition method based on KNN-DWDPC and random forest (RF) fusion is proposed. This method considers the influence of charging modes and designs intelligent charging pattern recognition for different vehicles.

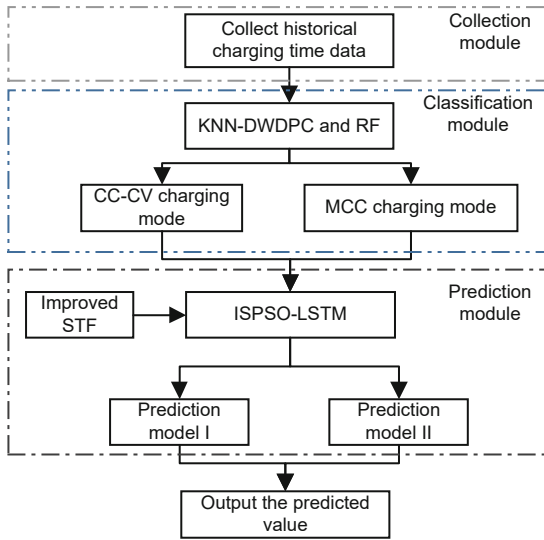


Fig. 4 Electric vehicle (EV) charging time prediction based on charging pattern recognition (RF: random forest; STF: strong tracking filter; CC-CV: constant current constant voltage; MCC: multistage constant current)

3. Focusing on the problem that PSO in a traditional PSO-LSTM algorithm can easily fall into a local optimum, an improved ISPSO method is proposed to determine the LSTM parameters. The method introduces the idea of using the mean value to improve the global optimal particle leadership.

4. Focusing on the problem that the convergence time of the algorithm is affected by the omission of the velocity term, the ISPSO-LSTM-STF charging time prediction method is proposed. This method can effectively improve the algorithm's prediction accuracy and reduce the convergence time.

3 Charging mode classification method based on KNN-DWDPC-RF

In this section, we introduce an intelligent model classification algorithm that can identify the characteristics of different charging modes, which is the prerequisite of the prediction in the next section.

The classification of charging modes is closely related to charging current. However, in a complete charging process, the variation of current data in the same charging mode is very different, which may lead to deterioration of classification and prediction effects. Therefore, before classification, we cluster all model data and classify charging modes after we cluster similar charging data into one data set. DPC

is used in this study. DPC is based on two important quantities: the local density ρ_i of each point i , and its distance δ_i from points of higher density. DPC takes the points with larger ρ_i and δ_i as the cluster center, and assigns the remaining data points to the cluster nearest to the center. Du et al. (2016) used the KNN idea to calculate the local density of DPC, and its calculation equation is as follows:

$$K(x_i) = \{j | j \in X, d(x_i, x_j) \leq d(x_i, NN_k(x_i))\}, \quad (8)$$

$$\rho_i = \exp\left(-\frac{\sum_{x_j \in K(x_i)} d(x_i, x_j)^2}{k}\right), \quad (9)$$

where d is the Euclidean distance and $NN_k(x_i)$ represents the k^{th} point nearest to x_i according to d . ρ_i calculates the average distance of k nearest data points to x_i , representing the inverse measure of distance. ρ_i uses the Euclidean distance as the measure of distance, and X is the data set.

In the DWDPC method, the Euclidean distance in local density (Eqs. (8) and (9)) is improved to the weighted Euclidean distance $d_w(x_i, x_j)$, which is calculated as follows:

$$\omega_{i,j,f} = \frac{x_{if} + x_{jf}}{\sum_{i,j=1}^n (x_{if} + x_{jf})}, \quad (10)$$

$$d_w(x_i, x_j) = \sqrt{\sum_{f=1}^m \omega_{i,j,f} (x_{if} - x_{jf})^2}, \quad (11)$$

where x_i and x_j represent the sample points to be calculated, x_{if} is the value of the f^{th} component of the i^{th} data point, n is the number of data points, m represents the sample dimension, and $\omega_{i,j,f}$ represents the distance weight.

The traditional Euclidean distance does not consider the distribution characteristics of data and cannot distinguish data attributes well. To increase the degree of distinction between data attributes, Wang et al. (2020) proposed the weighted Euclidean distance, but they considered only the distribution characteristics of the current point when calculating the distance between two points. Because the data distribution characteristics of relative points were not considered, the change of relative points was static in this method, and the weight of the distance between the same data points and different relative data points remained unchanged, which is obviously unreasonable. Compared with the existing methods and traditional Euclidean distance,

the proposed dynamic weighted Euclidean distance simultaneously considers the distribution characteristics of reference points and sample points in the sample, and different weights can be assigned.

The flowchart of the improved classification algorithm in this study is shown in Fig. 5.

The flow of KNN-DWDPC-RF is as follows:

Step 1: Import the historical charging data of various models from the charging pile database.

Step 2: Preprocess the data, including the completion and normalization of missing values. The input data of KNN-DWDPC-RF are obtained, consisting of charging current and charging SOC.

Step 3: Calculate $K(x_i)$, $\omega_{i,j,f}$, $d_w(x_i, x_j)$, δ_i , and ρ_i . δ_i represents the distance between a data point and the nearest point with a higher density. The cluster center distance δ_i of a data point is expressed as (Du et al., 2016)

$$\delta_i = \begin{cases} \min(d_w(x_i, x_j)), j : \rho_i > \rho_j, \\ \max(d_w(x_i, x_j)), \text{ otherwise.} \end{cases} \quad (12)$$

Step 4: Determine the cluster center.

Step 5: Sort the data points in descending order according to the density.

Step 6: Perform random sampling and determine the characteristics for each data set.

Step 7: Use the Gini coefficient to divide the property-generated nodes and establish the CART decision tree.

Step 8: Determine whether the number of decision trees is sufficient. If the requirements are met, random forest classifiers R_1, R_2, \dots, R_n are generated. If not, continue the iteration.

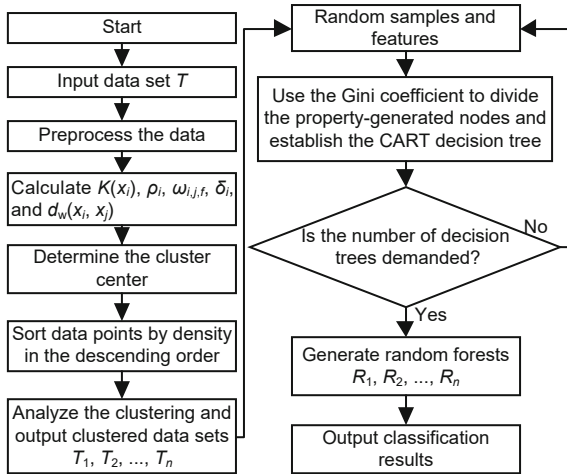


Fig. 5 Flowchart of KNN-DWDPC-RF

Step 9: Multiple random forest classifiers are used to classify multiple data sets, and the classification results are obtained.

Usually, there will be outlier data in the normal EV charging process. Compared with other works, the KNN-DWDPC method based on DPC, which can identify cluster classes and find outliers easily (Huang and Wu, 2005), is first used to cluster the charging data. It can improve the classification accuracy of RF and remove noise data, such as outliers in the data.

4 Charging time prediction method based on ISPSO-LSTM-STF

4.1 ISPSO algorithm

After the classification of charging modes, it is necessary to predict the charging time of the data. When LSTM is used to predict complex charging history data, setting algorithm parameters is very complicated, and the parameter quality directly affects the final prediction results. In this study, based on the traditional SPSO (Eq. (7)), a new ISPSO is proposed to determine the LSTM parameters, and ISPSO is proved to have better convergence effect in principle. The expression of ISPSO is as follows:

$$l_{id}^{k+1} = \omega l_{id}^k + c_1 r_1 \left(\frac{p_{id} + g_{bd}}{2} - l_{id}^k \right) + c_2 r_2 (g_{bd} - l_{id}^k), \quad (13)$$

where ω is the inertia weight, which can prevent the particle from falling into local optima. l_{id} is the position of the i^{th} particle in dimension, p_{id} is the current local optimal position of the particle, and g_{bd} is the global optimal position of the particle swarm. c_1 and c_2 are learning factors, and r_1 and r_2 are random numbers between 0 and 1.

Assuming that the particle positions of SPSO and ISPSO at the current time are the same, the comparison diagram of the updated positions of particles at the next time point is shown in Fig. 6. l_{id}^k and l'_{id}^k are the current positions of the particles, l_{id}^{k+1} is the updated position of SPSO, and l'_{id}^{k+1} is the update position of the new ISPSO. g_{bd} is the global optimal position and p_{id} is the local optimal position. It can be seen that the updated direction of particles is closer to the direction of the global optimal particle using the ISPSO method proposed in this study.

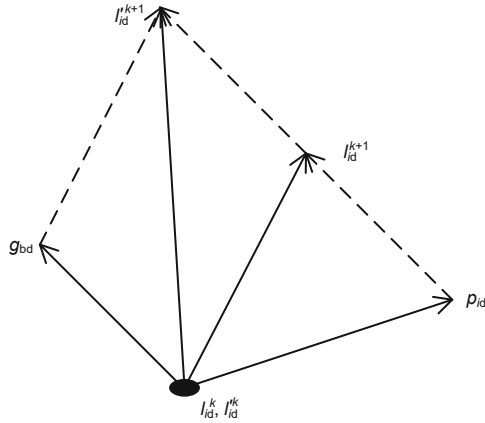


Fig. 6 Comparison diagram of position update

Theorem 1 When $\varphi < \omega < \varphi + 1$, ISPSO finally converges, and the convergence speed is higher than that of the SPSO method.

Proof Summarize Eq. (13):

$$l_{id}^{k+1} = (\omega - c_1 r_1 - c_2 r_2) l_{id}^k + c_1 r_1 \frac{p_{id} + g_{bd}}{2} + c_2 r_2 g_{bd}. \quad (14)$$

Define the parameters in the equation: $c_1 r_1 = \varphi_1$, $c_2 r_2 = \varphi_2$, $\varphi_1 + \varphi_2 = \varphi$, $(p_{id} + g_{bd})/2 = \beta_0$.

Put the defined parameters into Eq. (14) and sort them out to obtain Eq. (15):

$$\begin{aligned} l_{id}^{k+1} &= (\omega - \varphi) l_{id}^k + c_1 r_1 \beta_0 + c_2 r_2 g_{bd} \\ &= (\omega - \varphi) l_{id}^k + \varphi \beta, \end{aligned} \quad (15)$$

where $\beta = (c_1 r_1 \beta_0 + c_2 r_2 g_{bd})/\varphi$.

Eq. (15) can be written in the form of a differential equation (Hu and Li, 2007):

$$l(t+1) + (\varphi - \omega)l(t) = \varphi\beta. \quad (16)$$

The solution of the differential equation is

$$l(t) = C(\omega - \varphi)^t + \frac{\varphi\beta}{1 + \varphi - \omega},$$

where C is a constant.

To make $l(t)$ converge, when $T \rightarrow \infty$, $\left| \lim_{t \rightarrow \infty} l(t+1) / \lim_{t \rightarrow \infty} l(t) \right| < 1$ and $0 < \omega - \varphi < 1$.

Obtain the final solution: when $\varphi < \omega < \varphi + 1$,

$$\begin{aligned} \lim_{t \rightarrow \infty} l'(t) &= \frac{\varphi\beta}{1 + \varphi - \omega} \\ &= \frac{c_1 r_1 \beta_0 + c_2 r_2 g_{bd}}{1 + c_1 r_1 + c_2 r_2 - \omega} \\ &= \frac{c_1 r_1 (p_{id} + g_{bd})/2 + c_2 r_2 g_{bd}}{1 + c_1 r_1 + c_2 r_2 - \omega}. \end{aligned} \quad (17)$$

According to Eq. (17), the ISPSO algorithm converges to $\frac{c_1 r_1 (p_{id} + g_{bd})/2 + c_2 r_2 g_{bd}}{1 + c_1 r_1 + c_2 r_2 - \omega}$. Similarly, we can calculate the convergence value of the SPSO as

$$\lim_{t \rightarrow \infty} l(t) = \frac{c_1 r_1 p_{id} + c_2 r_2 g_{bd}}{1 + c_1 r_1 + c_2 r_2 - \omega}.$$

Suppose that the particle objective is to find the minimum position, and then $g_{bd} \leq p_{id}$, and

$$\frac{c_1 r_1 (p_{id} + g_{bd})/2 + c_2 r_2 g_{bd}}{1 + c_1 r_1 + c_2 r_2 - \omega} \leq \frac{c_1 r_1 p_{id} + c_2 r_2 g_{bd}}{1 + c_1 r_1 + c_2 r_2 - \omega}. \quad (18)$$

So, $\lim_{t \rightarrow \infty} l'(t) \leq \lim_{t \rightarrow \infty} l(t)$. Similarly, when the goal of the particle swarm is to find the maximum position, $\lim_{t \rightarrow \infty} l'(t) \geq \lim_{t \rightarrow \infty} l(t)$.

From the above proof, when $\varphi < \omega < \varphi + 1$, ISPSO finally converges, and the performance is better than that of SPSO. Therefore, ISPSO is a better LSTM parameter optimization method for charging time prediction.

4.2 STF-improved ISPSO-LSTM prediction method

The previous subsection proved that the ISPSO proposed in this study can approach the global optimal particle more accurately and has better convergence performance. Therefore, using ISPSO to optimize the parameters of the LSTM prediction model can improve the charging time prediction accuracy. Because the speed term is omitted, ISPSO is not limited by speed. So, a large amount of training data may lead to a local optimum at the later stage of training. To solve this problem, we use a strong tracking filter to improve the ISPSO position updating algorithm. After iteratively updating the position of each particle in the population, the algorithm will calculate their fitness and then judge whether the particle is good or bad by the fitness value. If the fitness of the particle's current location is better than that of the historical global optimum, the new location will replace the original global optimum and make the particle jump out of the local optimum, so that ISPSO can find the optimal parameter of LSTM and improve the prediction accuracy. Different from other improved PSO methods, we estimate the position and fitness of the particle rather than modeling and estimating the speed and position of the particle.

After the k^{th} iteration, the fitness value of a

particle at the x position is as follows:

$$h(\mathbf{x}) = \sum_{i=1}^n \|\mathbf{x}_i - \hat{\mathbf{x}}\|, \quad (19)$$

where $\hat{\mathbf{x}}$ is the mean value of \mathbf{x}_i ($i=1, 2, \dots, n$).

The filtering state equation is as follows (Li YY, 2020):

$$\mathbf{l}_k = \Phi_{k,k-1} \mathbf{l}_{k-1} + \mathbf{u}_{k,k-1} \mathbf{s}_{k,k-1}, \quad (20)$$

where \mathbf{l} is an n -dimensional state sequence representing the position state of each particle, $\Phi_{k,k-1}$ is the state transition matrix of the system, $\mathbf{u}_{k,k-1}$ is an n -dimensional input sequence, and \mathbf{s} is the sequence of process noise with $E(\mathbf{s}) = 0$ and $E(\mathbf{s}\mathbf{s}^T) = \mathbf{Q}_k$.

After each position update, the particle position is adjusted to improve the search efficiency. For each particle, after k iterations, the current fitness value of the particle can be expressed as follows (Li YY, 2020):

$$\mathbf{z}_k = h(\mathbf{l}_k) + \mathbf{v}_k, \quad (21)$$

where \mathbf{z}_k is the series of observations for the system, \mathbf{v}_k is the observation noise, and $E(\mathbf{v}) = 0$, $E(\mathbf{v}\mathbf{v}^T) = \mathbf{R}_k$. The strong tracking filter (STF) estimation model can be constructed according to Eqs. (20) and (21) (Li YY, 2020):

$$\hat{\mathbf{l}}_{k|k-1} = \Phi_{k,k-1} \hat{\mathbf{l}}_{k-1|k-1} + \mathbf{u}_{k,k-1}, \quad (22)$$

$$\hat{\mathbf{l}}_{k|k} = \hat{\mathbf{l}}_{k|k-1} + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{l}}_{k|k-1}), \quad (23)$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k)^{-1}, \quad (24)$$

$$\mathbf{P}_{k|k-1} = \lambda_k \Phi_{k,k-1} \mathbf{P}_{k-1|k-1} \Phi_{k,k-1}^T + \mathbf{Q}_{k,k-1}, \quad (25)$$

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1} (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^T, \quad (26)$$

$$\mathbf{N}_k = \mathbf{V}_k - \mathbf{H}_k \mathbf{Q}_{k,k-1} \mathbf{H}_k^T - \eta \mathbf{R}_k, \quad (27)$$

$$\mathbf{M}_k = \mathbf{H}_k \Phi_{k,k-1} \mathbf{P}_{k-1|k-1} \Phi_{k,k-1}^T \mathbf{H}_k^T, \quad (28)$$

$$\lambda_{0,k} \mathbf{M}_k = \mathbf{N}_k, \quad (29)$$

$$\lambda_k = \begin{cases} \lambda_{0,k}, & \lambda_{0,k} \geq 1, \\ 1, & \lambda_{0,k} < 1, \end{cases} \quad (30)$$

where $\mathbf{H}_k = \left. \frac{\partial h(\mathbf{l}_k)}{\partial \mathbf{l}_k} \right|_{\mathbf{l}_k = \hat{\mathbf{l}}_{k|k-1}}$, η is the weakening factor whose function is to reduce the possibility of overregulation, λ_k is the fading factor which can effectively suppress the divergence of the filter, $\hat{\mathbf{l}}_{k|k}$ is the estimate of the state, k is the moment, \mathbf{K}_k

is the gain matrix, and $\mathbf{P}_{k|k}$ is the prediction error covariance matrix.

After each iteration, the particle position estimated by the filter is calculated and compared with the fitness of the particles in the population. If the fitness value of the position estimated by the STF estimation model is better than the values of the global best and local best of the current population, the estimated model particle position is used instead. \mathbf{V}_k is the new information covariance matrix, and the calculation equation is as follows (Xu et al., 2019):

$$\mathbf{V}_k = \begin{cases} \gamma_1 \gamma_1^T, & k = 0, \\ \frac{\rho \mathbf{V}_{k-1} + \gamma_k \gamma_k^T}{1 + \rho}, & k \geq 1, \end{cases} \quad (31)$$

$$\gamma_k = \mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{l}}_{k|k-1}, \quad (32)$$

where ρ ($0 < \rho < 1$) represents the forgetting factor.

The algorithm flowchart is given in Fig. 7.

In this study, the steps of the charging time prediction method of ISPSO-LSTM-STF are as follows:

Step 1: Import the historical charging data of different models from the charging pile database.

Step 2: Preprocess data, including normalization and completion of missing values.

Step 3: Initialize the ISPSO algorithm parameters and strong tracking filter parameters, including the population size, the number of iterations, and the limited interval of the particle position.

Step 4: Use STF to estimate the particle position and fitness value of ISPSO. The number of neurons (h_1 and h_2 for each layer) for LSTM, the learning rate r , and the number of iterations n are taken as the optimization objects of the particle.

Step 5: Compare the fitness values of the local best advantage of ISPSO and the local best advantage estimated by a strong tracking filter. The particle with the smallest fitness value is selected as the best local advantage.

Step 6: Train the LSTM model with the optimal parameters obtained.

Step 7: Use the trained LSTM model to predict the real charging data set.

5 Experiments and analysis

5.1 Experimental data

The algorithms used in this study are all run in the MATLAB 2019b programming environment.

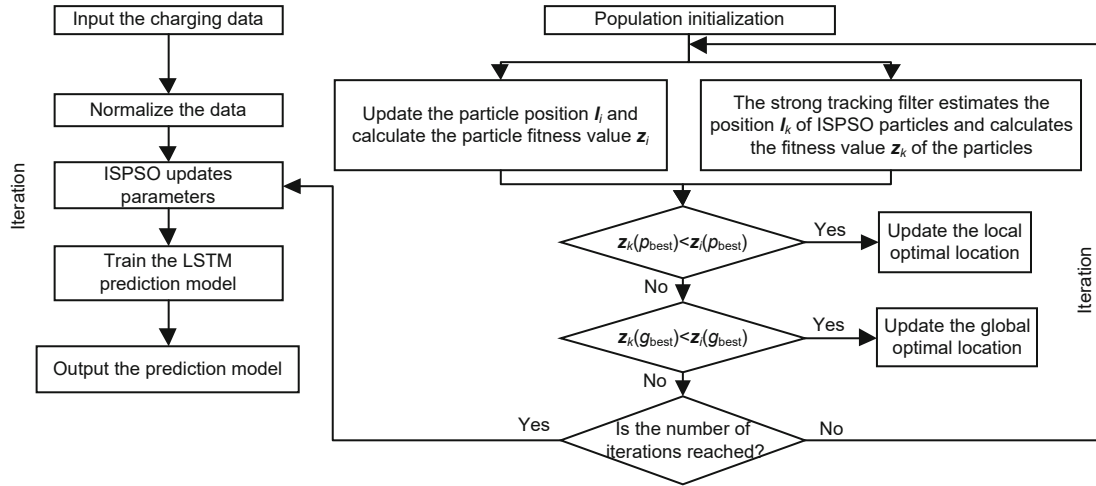


Fig. 7 Flowchart of ISPSO-LSTM-STF

All EV data used in this work came from a charging point operating company in Shenzhen. So far, the company has accumulated >3 million charging instances. The main charging pile types are Win-line Technology YLCE-120750 and INFY POWER IDS120K750D-X5, both following the national standard: GB/T 27930-2015 Communication Protocols Between Off-board Conductive Charger and Battery Management System. The charging pile transmits the message to the company's charging safety monitoring platform through a controller area network (CAN) bus and Ethernet, and saves the data in a time-series database.

To minimize the impact of external temperature, the vehicles selected in this study are all continuously charged in the same time period every day. The interval for each packet is about 5–10 s. Due to the time interval during packet transmission, several packets may belong to the same SOC. Therefore, we average the voltage and current values of the corresponding messages of each SOC, and obtain the corresponding SOC and the corresponding electric quantity, average current, and average voltage, which are used as data input.

However, in data transmission, there may be false positives, missing positives, delays, and so on. Therefore, missing values and outliers in the charging data set need to be identified and processed. In this study, we consider that the charging data that exceed the upper limit of charging power are incorrect and can be deleted. Finally, the interpolation method is

used to fill in the deleted and missing data.

The min-max normalization method is applied to charging data in this experiment (Ge et al., 2022).

5.2 Experimental results and analysis of the classification model

Charging current and SOC are used in charging mode classification. The charging data randomly extracted from the company database are divided into seven groups, and each group contains 299 test samples and 299 training samples. Every group is two-dimensional data composed of charging time and charging current. The parameters are set as follows: $k = 15$, the number of decision trees is 150, and the number of iterations is 250. To verify the effectiveness of the improved classification algorithm, the algorithm is compared with RF and DPC-RF. Classification accuracy P is used to evaluate the effect of mode classification (Li B et al., 2020):

$$P = \frac{TP + TN}{TP + FP + TN + FN}. \quad (33)$$

TP is the correct number of CC-CV samples, TN is the correct number of MCC samples, FP is the number of classification errors in CC-CV samples, and FN is the number of classification errors of MCC samples. The experimental results classified in this study are shown in Fig. 8. It can be clearly seen that clustering and then classifying the data can effectively reduce the classification error. In addition, among the seven data groups, the accuracy of

KNN-DWDPC-RF is better than those of DPC-RF and RF, and the classification effect is more stable.

5.3 Experimental results and analysis of the prediction model

SOC, charging voltage, charging current, and electric quantity are used to predict charging time.

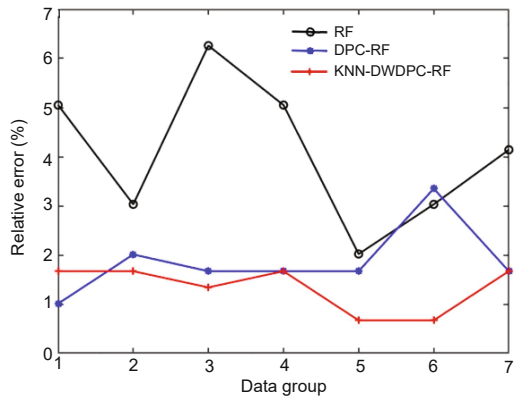


Fig. 8 Comparison of accuracy under different classification methods

Parameters are set as follows: particle swarm inertia weight $\omega = 0.9$, learning factor $c_1 = 2$, $c_2 = 2$, numbers of iterations of ISPSO and PSO are both 100, and the population size is 20. In LSTM, the numbers of two hidden layers are $h_1, h_2 \in [1, 200]$, the learning rate is $r \in [0.001, 0.01]$, and the number of training times is $n \in [1, 500]$. In strong trace filtering, $Q = 0.001$, $R = 10$.

Four vehicles with two CC-CV and two MCC charging modes are selected in this study. Their SOC changes are all $>60\%$ during charging to make results more credible.

The prediction algorithm in this study uses the relative error (RE), absolute error (AE), and mean absolute percentage error (MAPE) as evaluation indexes in the experiments (Ge et al., 2022).

1. Vehicle 1

Vehicle 1 is a vehicle in MCC charging mode. Fig. 9a is the LSTM charging time prediction without charging mode classification. Figs. 9b–9e are the charging time predictions that classify charging mode first and then forecast using LSTM,

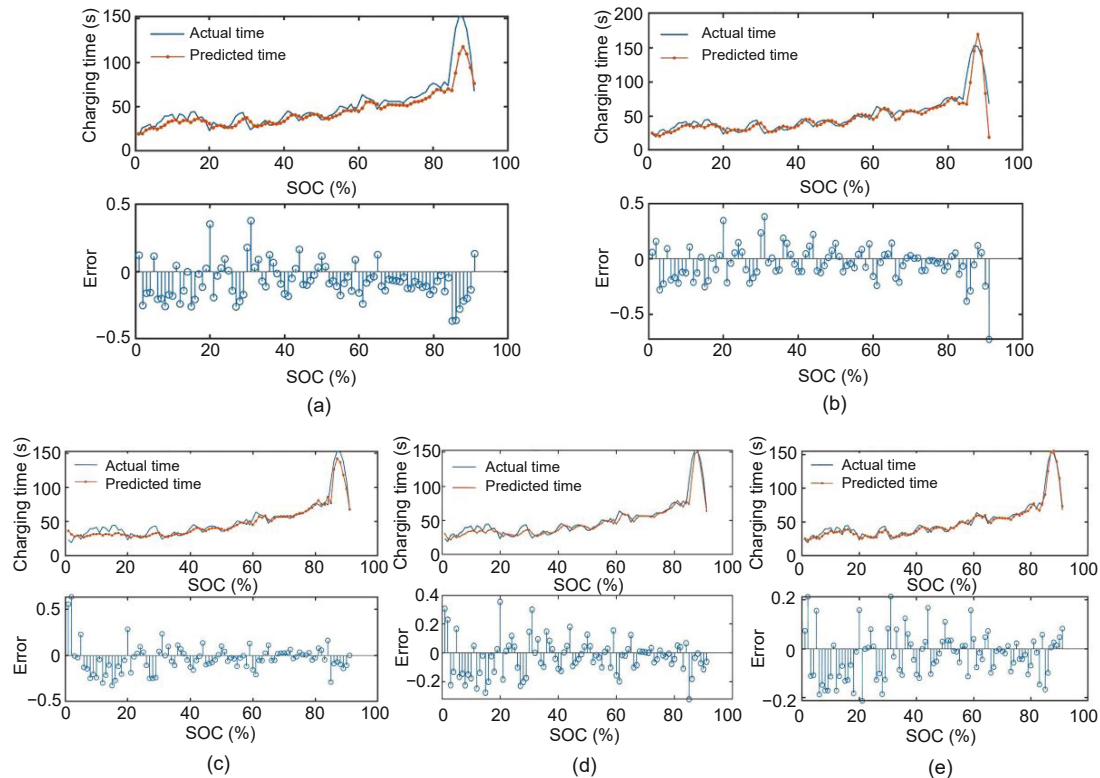


Fig. 9 Charging time prediction of vehicle 1 under different methods: (a) LSTM without mode classification; (b) LSTM with mode classification; (c) PSO-LSTM with mode classification; (d) ISPSO-LSTM with mode classification; (e) ISPSO-LSTM-STF with mode classification (SOC: state of charge)

PSO-LSTM, ISPSO-LSTM, and ISPSO-LSTM-STF methods, respectively. As shown in Fig. 9, charging mode classification can effectively improve the accuracy when the charging time has a sudden change. Compared with ISPSO-LSTM and PSO-LSTM, ISPSO-LSTM-STF is more accurate in the early and late stages of charging.

As can be seen from Table 1, the real full charging time of vehicle 1 is 76.0851 min, and the predicted values of the full charging time are 68.6775 min for LSTM (unclassified), 71.2542 min for LSTM (classified), 72.1804 min for PSO-LSTM, 72.5088 min for ISPSO-LSTM, and 73.7123 min for ISPSO-LSTM-STF. The corresponding REs are

9.74%, 6.35%, 5.14%, 4.70%, and 3.12%. The error after mode classification is smaller than that without classification. The RE of ISPSO-LSTM-STF is the smallest.

2. Vehicle 2

Vehicle 2 is a vehicle in MCC charging mode. Fig. 10a is the LSTM method without charging mode classification. Figs. 10b–10e are the charging time predictions that classify charging mode first and then forecast using LSTM, PSO-LSTM, ISPSO-LSTM, and ISPSO-LSTM-STF methods, respectively.

In Table 2, the prediction of charging time in the unclassified model has a large error in the overall prediction time. After charging mode classification,

Table 1 Comparison of relative errors of vehicle 1 under different methods

Method	Actual charging time (min)	Predicted charging time (min)	Relative error (%)
LSTM (unclassified)	76.0851	68.6775	9.74
LSTM (classified)	76.0851	71.2542	6.35
PSO-LSTM	76.0851	72.1804	5.14
ISPSO-LSTM	76.0851	72.5088	4.70
ISPSO-LSTM-STF	76.0851	73.7123	3.12

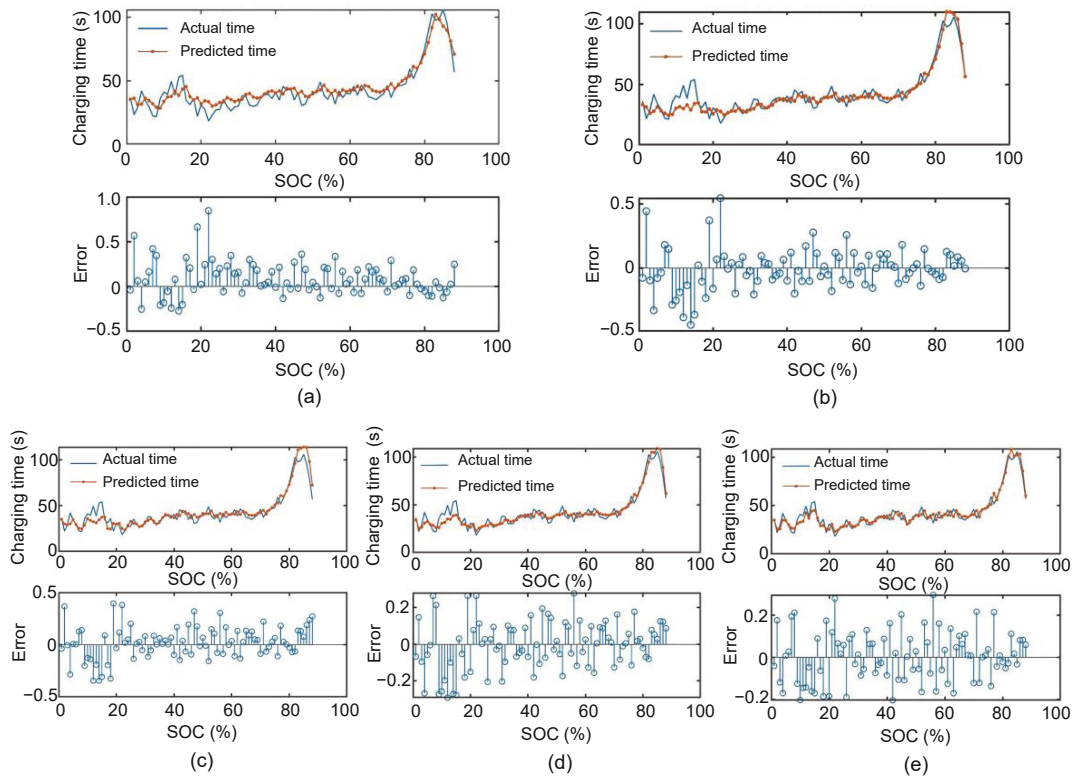


Fig. 10 Charging time prediction of vehicle 2 under different methods: (a) LSTM without mode classification; (b) LSTM with mode classification; (c) PSO-LSTM with mode classification; (d) ISPSO-LSTM with mode classification; (e) ISPSO-LSTM-STF with mode classification (SOC: state of charge)

the accuracy of medium-term charging prediction is improved, but for the abrupt change data, the prediction error is still large. Compared with other methods, ISPSO-LSTM-STF performs best in the face of data mutation. Although the RE in the early stage is large, the charging time in the early stage is generally only a few seconds or tens of seconds.

3. Vehicle 3

Vehicle 3 is a vehicle in CC-CV charging mode. Fig. 11a is the LSTM without charging mode classification. Figs. 11b–11e are the charging time predictions that classify charging mode first and then forecast using LSTM, PSO-LSTM, ISPSO-LSTM, and ISPSO-LSTM-STF methods, respectively.

As can be seen from Table 3, the real full charging time of vehicle 3 is 83.9672 min; the predicted values of the full charging time are 75.7101 min for LSTM (unclassified), 80.5296 min for LSTM (classified), 81.7740 min for PSO-LSTM, 85.9869 min for ISPSO-LSTM, and 83.9812 min for ISPSO-LSTM-STF. Accordingly, the REs are 9.83%, 4.09%, 2.61%, 2.41%, and 0.02%, respectively.

4. Vehicle 4

Vehicle 4 is a vehicle in CC-CV charging mode. Fig. 12a is the LSTM charging time prediction of vehicle 4 without charging mode classification. Figs. 12b–12e are the charging time predictions that classify charging mode first and then forecast

Table 2 Comparison of relative errors of vehicle 2 under different methods

Method	Actual charging time (min)	Predicted charging time (min)	Relative error (%)
LSTM (unclassified)	63.0132	66.0396	4.80
LSTM (classified)	63.0132	61.3405	2.65
PSO-LSTM	63.0132	63.8580	1.34
ISPSO-LSTM	63.0132	62.2409	1.23
ISPSO-LSTM-STF	63.0132	62.6676	0.55

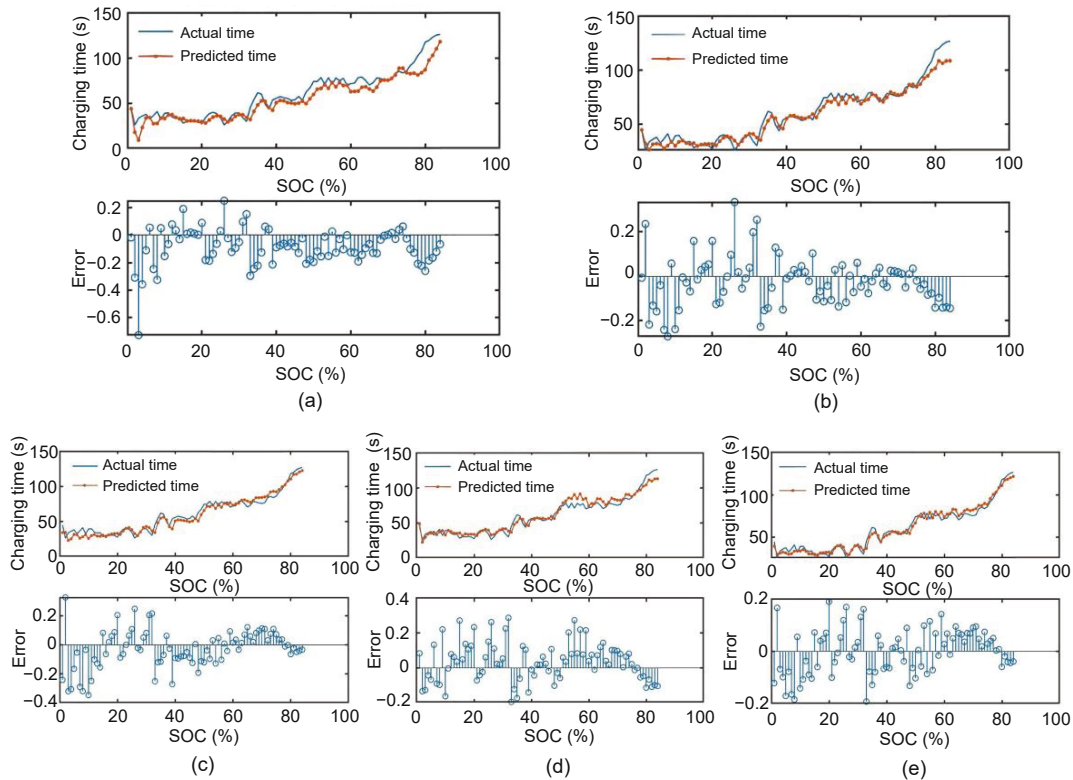


Fig. 11 Charging time prediction of vehicle 3 under different methods: (a) LSTM without mode classification; (b) LSTM with mode classification; (c) PSO-LSTM with mode classification; (d) ISPSO-LSTM with mode classification; (e) ISPSO-LSTM-STF with mode classification (SOC: state of charge)

using LSTM, PSO-LSTM, ISPSO-LSTM, and ISPSO-LSTM-STF methods, respectively.

In Table 4, the real full charging time of vehicle 4 is 54.8786 min, and the predicted values of the full charging time are 49.8475 min for LSTM (unclassified), 51.9700 min for LSTM (classified), 52.1182 min for PSO-LSTM, 55.7403 min for ISPSO-LSTM, and

54.9682 min for ISPSO-LSTM-STF. Corresponding REs are 9.17%, 5.30%, 5.03%, 1.57%, and 0.16%, respectively.

Table 5 summarizes the MAPE of the four electric vehicles under different methods. It can be seen that in two different charging modes, the prediction effect of charging data after classification is far

Table 3 Comparison of relative errors of vehicle 3 under different methods

Method	Actual charging time (min)	Predicted charging time (min)	Relative error (%)
LSTM (unclassified)	83.9672	75.7101	9.83
LSTM (classified)	83.9672	80.5296	4.09
PSO-LSTM	83.9672	81.7740	2.61
ISPSO-LSTM	83.9672	85.9869	2.41
ISPSO-LSTM-STF	83.9672	83.9812	0.02

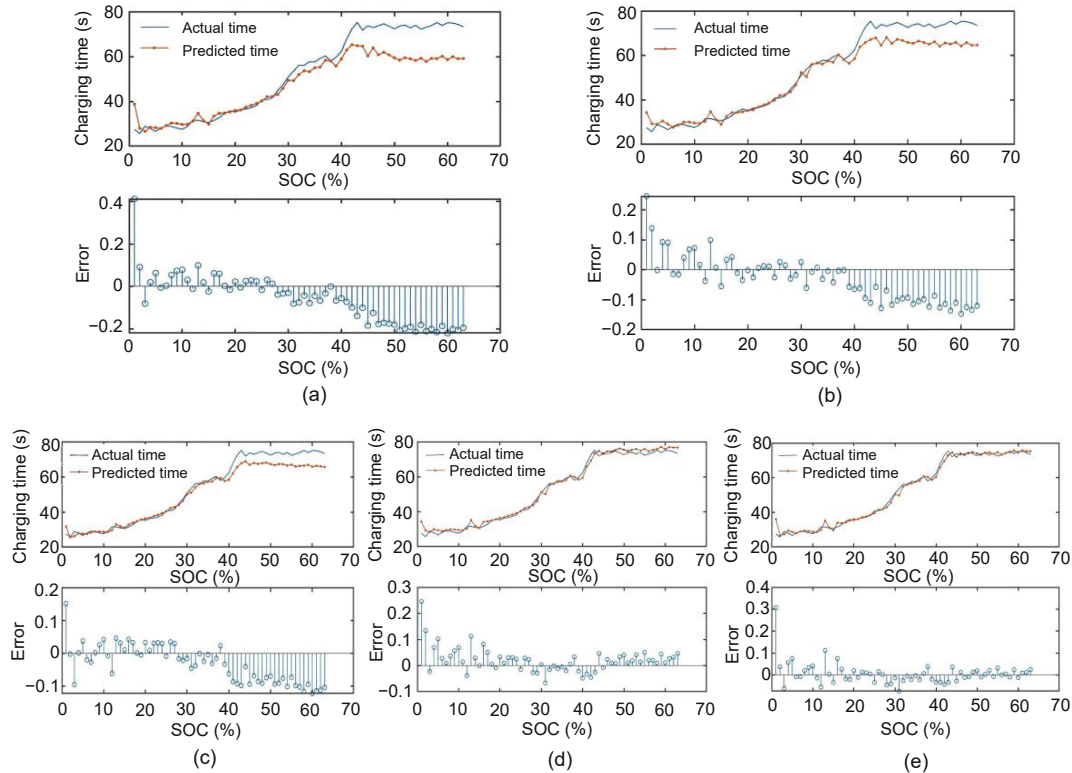


Fig. 12 Charging time prediction of vehicle 4 under different methods: (a) LSTM without mode classification; (b) LSTM with mode classification; (c) PSO-LSTM with mode classification; (d) ISPSO-LSTM with mode classification; (e) ISPSO-LSTM-STF with mode classification (SOC: state of charge)

Table 4 Comparison of relative errors of vehicle 4 under different methods

Method	Actual charging time (min)	Predicted charging time (min)	Relative error (%)
LSTM (unclassified)	54.8786	49.8475	9.17
LSTM (classified)	54.8786	51.9700	5.30
PSO-LSTM	54.8786	52.1182	5.03
ISPSO-LSTM	54.8786	55.7403	1.57
ISPSO-LSTM-STF	54.8786	54.9682	0.16

better than that without classification. The proposed ISPSO-LSTM has a slightly better result than the PSO-improved LSTM. Meanwhile, the mean absolute percentage error of ISPSO-LSTM-STF has been reduced by 61% compared to that of ISPSO-LSTM.

Table 5 Mean absolute percentage error (MAPE) under different methods

Method	MAPE (%)
LSTM (unclassified)	8.385
LSTM (classified)	4.598
PSO-LSTM	3.530
ISPSO-LSTM	2.478
ISPSO-LSTM-STF	0.963

6 Conclusions

In this paper, a novel charging time prediction method, i.e., ISPSO-LSTM-STF, has been proposed to solve the problem of low accuracy encountered by traditional methods due to the lack of charging mode recognition. First, an improved DWDPC method based on KNN has been used to classify EV charging modes according to charging voltage and current. Then charging time has been predicted by an LSTM algorithm whose parameters were optimized by ISPSO and STF. It has been proved by experiments that this ISPSO-LSTM-STF method, based on charging pattern recognition, can effectively improve the charging time prediction accuracy and has real engineering value.

Contributors

Chunxi LI designed the research and gave the main frame. Chunxi LI and Yingying FU processed the data. Yingying FU tested the MATLAB program and drafted the paper. Xiangke CUI and Quanbo GE helped organize the paper and gave a lot of suggestions. Chunxi LI, Xiangke CUI, and Quanbo GE revised and finalized the paper.

Compliance with ethics guidelines

Chunxi LI, Yingying FU, Xiangke CUI, and Quanbo GE declare that they have no conflict of interest.

Data availability

Data are not available due to commercial restrictions.

References

- Alché F, de la Fortelle A, 2017. An LSTM network for highway trajectory prediction. *Proc IEEE 20th Int Conf on Intelligent Transportation Systems*, p.353-359. <https://doi.org/10.1109/ITSC.2017.8317913>
- Chang XQ, Song ZX, Wang JH, 2020. Electric vehicle charging load prediction and system development based on Monte Carlo algorithm. *High Volt Appar*, 56(8):1-5 (in Chinese). <https://doi.org/10.13296/j.1001-1609.hva.2020.08.001>
- Cheng SY, Lin PC, Lin PJ, 2019. New method of predict remaining charging time for lithium-ion batteries. *Chin J Power Sources*, 43(1):99-102, 135 (in Chinese). <https://doi.org/10.3969/j.issn.1002-087X.2019.01.028>
- Du MJ, Ding SF, Jia HJ, 2016. Study on density peaks clustering based on k -nearest neighbors and principal component analysis. *Knowl-Based Syst*, 99:135-145. <https://doi.org/10.1016/j.knosys.2016.02.001>
- Frendo O, Graf J, Gaertner N, et al., 2020. Data-driven smart charging for heterogeneous electric vehicle fleets. *Energy AI*, 1:100007. <https://doi.org/10.1016/j.egyai.2020.100007>
- Ge QB, Guo C, Jiang HY, et al., 2022. Industrial power load forecasting method based on reinforcement learning and PSO-LSSVM. *IEEE Trans Cybern*, 52(2):1112-1124. <https://doi.org/10.1109/TCYB.2020.2983871>
- Han T, Xu M, Lu YF, 2014. Charging Remaining Time Estimation Method, Device and Mobile Equipment. CN Patent 201210297383.4 (in Chinese).
- Hu W, Li ZS, 2007. A simpler and more effective particle swarm optimization algorithm. *J Softw*, 18(4):861-868 (in Chinese). <https://doi.org/10.1360/jos180861>
- Huang TS, Wu SY, 2005. Enhanced particle swarm optimization algorithm based on Kalman filter principle. *Comput Eng Appl*, 41(35):56-58 (in Chinese). <https://doi.org/10.3321/j.issn:1002-8331.2005.35.019>
- Li B, Han R, He YG, et al., 2020. Applications of the improved random forest algorithm in fault diagnosis of motor bearings. *Proc CSEE*, 40(4):1310-1319 (in Chinese). <https://doi.org/10.13334/j.0258-8013.pcsee.190501>
- Li CX, Fu YY, Zhang JM, et al., 2020. Evaluation methods on charging safety for EV power battery. *Proc 35th Youth Academic Annual Conf of Chinese Association of Automation*, p.318-323. <https://doi.org/10.1109/YAC51587.2020.9337499>
- Li YY, 2020. Engineering Intelligent Kalman Filtering Method. MS Thesis, Hangzhou Dianzi University, Hangzhou, China (in Chinese). <https://doi.org/10.27075/d.cnki.ghzdc.2020.000227>
- Lin PC, 2018. Tri-section SVR model for predicting charging time of lithium-ion batteries. *Chin J Power Sources*, 42(8):1155-1157, 1232 (in Chinese). <https://doi.org/10.3969/j.issn.1002-087X.2018.08.019>
- Liu B, 2020. Research of Short-Term Power Load Forecasting Based on PSO-LSTM Algorithm. MS Thesis, Jilin University, Jilin, China (in Chinese). <https://doi.org/10.27162/d.cnki.gjlin.2020.003369>
- Liu WW, Sang SB, Zhang HP, 2022. Study on improved heart sound classification model based on CNN+LSTM. *Electron Des Eng*, 30(2):38-42 (in Chinese). <https://doi.org/10.14022/j.issn1674-6236.2022.02.009>

- Liu X, 2020. Prediction of Lithium Battery's Remaining Charging Time Based on IndyLSTM. MS Thesis, Xi'an University of Science and Technology, Xi'an, China (in Chinese).
<https://doi.org/10.27397/d.cnki.gxaku.2020.000677>
- Liu YY, 2018. Research on modeling and simulation of detailed model of photovoltaic power generation system. *Comput Knowl Technol*, 14(10):221-224, 228 (in Chinese). <https://doi.org/10.14004/j.cnki.ckt.2018.1153>
- Rodriguez A, Laio A, 2014. Clustering by fast search and find of density peaks. *Science*, 344(6191):1492-1496. <https://doi.org/10.1126/science.1242072>
- Roondiwala M, Patel H, Varma S, 2017. Predicting stock prices using LSTM. *Int J Sci Res*, 6(4):1754-1756.
- Sun J, Chen SB, You PC, et al., 2021. Battery-assisted online operation of distributed data centers with uncertain workload and electricity prices. *IEEE Trans Cloud Comput*, early access.
<https://doi.org/10.1109/TCC.2021.3132174>
- Wang ZL, Li J, Song YF, 2020. Improved K -means algorithm based on distance and weight. *Comput Eng Appl*, 56(23):87-94 (in Chinese).
<https://doi.org/10.3778/j.issn.1002-8331.2009-0103>
- Xu ZY, Kang Y, Cao Y, et al., 2019. Man-machine verification of mouse trajectory based on the random forest model. *Front Inform Technol Electron Eng*, 20(7):925-929. <https://doi.org/10.1631/FITEE.1700442>
- Yang QM, Liu GL, Bao YN, et al., 2022. Fault detection of wind turbine generator bearing using attention-based neural networks and voting-based strategy. *IEEE/ASME Trans Mechatron*, 27(5):3008-3018.
<https://doi.org/10.1109/TMECH.2021.3127213>
- Zhang QS, Zhao QC, 2020. Effects of overcharge cycling on the aging and safety of lithium ion batteries. *High Volt Eng*, 46(10):3390-3397 (in Chinese).
<https://doi.org/10.13336/j.1003-6520.hve.20200361>
- Zhang YF, Zhao ZD, Deng YJ, et al., 2021. ECGID: a human identification method based on adaptive particle swarm optimization and the bidirectional LSTM model. *Front Inform Technol Electron Eng*, 22(12):1641-1654.
<https://doi.org/10.1631/FITEE.2000511>
- Zhou D, Song XH, Lu WB, et al., 2019. Real-time SOH estimation algorithm for lithium-ion batteries based on daily segment charging data. *Proc CSEE*, 39(1):105-111 (in Chinese).
<https://doi.org/10.13334/j.0258-8013.pcsee.181026>
- Zhu XQ, Wang ZP, Wang C, et al., 2019. An experimental study on overcharge behaviors of lithium-ion power battery with $\text{LiNi}_{0.6}\text{Co}_{0.2}\text{Mn}_{0.2}\text{O}_2$ cathode. *Automot Eng*, 41(5):582-589 (in Chinese).
<https://doi.org/10.19562/j.chinasae.qcgc.2019.05.015>
- Zhu ZC, Zheng YJ, 2017. Prediction method of rechargeable electricity of vehicle battery at different temperature. *Agric Equip Veh Eng*, 55(8):1-5.
<https://doi.org/10.3969/j.issn.1673-3142.2017.08.001>