



Optimised LSTM Neural Network for Traffic Speed Prediction with Multi-Source Data Fusion

Yongpeng ZHAO¹, Yongcang LI², Changxi MA³, Ke WANG⁴, Xuecai XU⁵

Original Scientific Paper
Submitted: 8 Jan. 2024
Accepted: 3 June 2024

¹ 13210007@stu.lzjtu.edu.cn, Lanzhou Jiaotong University, School of Traffic and Transportation; Gansu Provincial Highway and Transportation Construction Group Co., Ltd

² 823833852@qq.com, Gansu Road and Bridge Construction Group Co., Ltd

³ machangxi@mail.lzjtu.cn, Lanzhou Jiaotong University, School of Traffic and Transportation

⁴ 2232308339@qq.com, Lanzhou Jiaotong University, School of Traffic and Transportation

⁵ Corresponding author, xuecai_xu@hust.edu.cn, Huazhong University of Science and Technology, School of Civil and Hydraulic Engineering



This work is licensed under a Creative Commons Attribution 4.0 International License.

Publisher:
Faculty of Transport and Traffic Sciences,
University of Zagreb

ABSTRACT

Predicting traffic speed accurately and in real-time is crucial for the development of smart transportation systems. Given the nonlinear and stochastic nature of vehicle data, integrating diverse spatio-temporal data sources with the Improved Particle Swarm Optimisation (IPSO) offers a promising approach to optimise the Long Short-Term Memory Neural Network (LSTM). Firstly, we enhance the optimisation capabilities of PSO by implementing nonlinear inertial weight and adaptive variation. Secondly, addressing the challenge of selecting the LSTM hyperparameters, the PSO algorithm effectively identifies global optimal solutions for hyperparameter optimisation, ensuring appropriate settings through iterative training. Subsequently, we conduct a case study using multi-source spatio-temporal traffic speed data, comparing our proposed IPSO-LSTM model with traditional neural network prediction models and advanced models. Results from the experiment demonstrate that the IPSO-LSTM model presented in this study addresses issues of parameter selection and inaccurate prediction encountered by traditional LSTM models in traffic state prediction. Moreover, it enhances the model's ability to capture speed time series dynamics. Notably, in processing complex speed data, our model exhibits superior accuracy and stability in prediction.

KEYWORDS

traffic speed prediction; multi-source data fusion; improved particle swarm optimisation; long short-term memory network.

1. INTRODUCTION

In recent years, with the steady increase in the number of automobiles on roads, traffic congestion has escalated significantly. Accurate and real-time prediction of vehicle speed plays a pivotal role in travellers' route planning and time management endeavours. This predictive capability not only mitigates congestion during peak hours but also furnishes crucial support for traffic management authorities in devising proactive strategies. Accurately anticipating road speeds facilitates pre-emptive measures to circumvent congestion before it materialises. Hence, the development of speed prediction algorithms utilising vehicle driving data carries profound practical implications for the sophisticated control of highway vehicles.

In the domain of automotive intelligence, endeavours such as vehicle safety-assisted driving and intelligent vehicle behaviour decision analysis necessitate the examination of vehicle driving data to formulate corresponding strategies. The timelier and more precise the data acquisition, the higher the quality of ensuing decisions. Through speed prediction algorithms, driving data for future periods can be relayed to vehicle decision-making entities for analysis, thereby enabling the formulation of optimal behavioural strategies. Consequently, speed prediction algorithms emerge as indispensable components of vehicle intelligence.

Intelligent Transportation Systems (ITS) stand to enhance performance by assimilating future road network data. Effective traffic prediction requires the comprehension of the non-linear relationship between past and future traffic conditions. This functionality confers benefits across various applications, encompassing route guidance and congestion management [1]. This study endeavours to augment traffic prediction accuracy and minimise network operation costs, thereby furnishing insights pertinent to traffic management and control.

The remainder of the paper is structured as follows: Section 2 provides a comprehensive review of the pertinent literature. Section 3 elucidates the prediction model based on deep learning, the enhanced particle swarm optimisation algorithm, and delineates the algorithmic steps in detail. Section 4 elaborates on the data format and processing procedures. Section 5 undertakes numerical calculations and conducts an analysis of the results. Finally, in Section 6, the research findings are summarised, along with an examination of limitations and potential avenues for future research.

2. LITERATURE REVIEW

There are two standard methods for predicting traffic conditions. One category involves parametric approaches such as autoregressive integrated moving average (ARIMA), multiple regression theory, exponential smoothing, grey model and Markov model, among others. The second category encompasses non-parametric methods such as the k-nearest neighbour model, neural networks, and more.

Among them, the parametric method helps elucidate the quantitative relationship between independent and dependent variables, and quantitatively analyse the influence of variables on dependent variables. Chandra and Al-Deek considered the correlation of traffic data (especially speed) between different locations, and introduced the VAR model as the prediction model of expressway section speed [2]. Sun and Sun introduced a real-time collision prediction model employing a dynamic Bayesian network (DBN). Their goal was to analyse the relationship between dynamic traffic changes and collisions, specifically using expressway traffic speed data for real-time accident prediction [3]. Wang et al. merged empirical mode decomposition (EMD) with autoregressive integrated moving average to forecast and analyse non-stationary and nonlinear traffic speed data series [4]. Zhang et al. developed a traffic speed prediction framework employing high-order Markov chains. This framework encompassed both observable factors like traffic volume and speed, as well as latent environmental factors [5]. Zeng et al. introduced the DWT-Bi-LSTM model, leveraging wavelet transform and bidirectional LSTM, to enhance accuracy and efficiency in predicting parking spot availability from past data [6]. Ma et al. introduced a novel wavelet-based spatiotemporal multi-graph convolutional neural network to improve the precision of traffic speed prediction. By integrating multiple GCNs, 3DCNN and wavelet analysis, their model exhibited superior performance compared to standard methods across diverse time spans and real-world datasets. [7].

Non-parametric methods adapt internal model parameters based on data to establish relationships between variables and dependent variables. They are less influenced by user intervention, thus exhibiting broader applicability. Research by Asif et al. and Yao et al. proposed a traffic speed prediction model with single time step length that comprehensively considered spatio-temporal parameters on the basis of the support vector machine (SVM) algorithm, so as to support travellers' route selection and traffic guidance [8–9]. Tang et al. proposed a new fuzzy artificial neural network structure to predict multi-step forward driving speed based on three remote traffic microwave sensors in the southern section of the fourth Ring road in Beijing [10]. Zang et al. introduced an MSTFLN model designed for long-term expressway traffic speed prediction. This model extracts intricate speed features by jointly learning from multi-scale inputs [11]. Zheng et al. proposed a k-nearest neighbour (K-NN) method based on tensor, in which traffic mode involved multidimensional time information and bidirectional spatial information [12]. Zhang et al. adjusted the hyperparameters of the multi-task learning (MTL) model by using the Bayesian optimisation method in combination with GPS data of a taxi, and concluded that the MTL model is leveraging deep learning [13]. Zhang et al. and Zheng et al. introduced the attention-graph convolution sequence-to-sequence model (AGC-SEQ2SEQ), a deep learning framework aimed at capturing intricate non-stationary temporal dynamics

and spatial correlations for multi-step traffic condition prediction [14–15]. Ma et al. devised a short-term traffic speed prediction technique utilising a hybrid spatio-temporal feature selection algorithm (STFSA), which integrates traffic flow spatio-temporal analysis. They combined convolutional neural network and gated recursive unit (CNN-GRU) methodologies for this purpose [16]. Ma et al. introduced swarm intelligence optimised deep extreme learning machines (DELM) and applied them, along with variational mode decomposition (VMD), to road traffic prediction and autonomous lane change decisions [17]. While neural networks offer various advantages, their predictive capacity diminishes when handling long time series data, thus prompting the introduction of the Long Short-Term Memory (LSTM) neural network.

In order to verify the effectiveness of the LSTM neural network in predicting traffic speed, Ma et al. employed the travel speed data of Beijing traffic microwave detector to conduct model training and testing [18]. Jia et al. assessed the efficacy of both deep belief networks (DBN) and the LSTM models in short-term traffic speed prediction. Findings indicated the LSTM's superior capability in capturing the temporal features of traffic speed data compared to DBN [19]. Gu et al. fused the LSTM with gated cyclic unit (GRU) neural networks to create a two-layer deep learning framework (FDL) [20]. Hu et al. incorporated the LSTM to address the challenge of gradient disappearance in practical applications, integrated the attention mechanism into the LSTM-RNN, and devised a high-precision short-term traffic flow prediction model [21]. Meng et al. innovatively introduced a dynamic time distortion algorithm for time series processing and proposed a long short-term memory (D-LSTM) model with dynamic time distortion [22]. Zeng et al. proposed a hybrid GRU-LSTM model using historical parking data and various factors to forecast parking availability, aiming to optimise resource use and ease traffic congestion amid growing urbanisation and vehicle numbers [23]. Ma et al. introduced an improved short-term traffic flow prediction model that combines time series analysis with an enhanced LSTM network, specifically leveraging bidirectional LSTM (BiLSTM) [24]. While neural networks offer various advantages, their ability to predict over extended time series is limited. This led to the proposal of the Long Short-Term Memory (LSTM) neural network. Despite the LSTM's strong performance, enhancing its predictive accuracy remains a concern due to challenges in determining the optimal number of hidden layers and neurons within those layers. Additionally, setting the appropriate learning rate and iteration count poses difficulties. The number of hidden layers and neurons significantly impacts the model's fitting ability, whereas the learning rate and iteration count influence the training process and the model's effectiveness. In practical applications, determining these parameters relies on empirical experience, resulting in considerable randomness that diminishes the predictive capability of the model.

Moreover, within intricate urban road networks, the traffic state is subject to numerous interrelated factors that exhibit complex nonlinear relationships with speed. Among them, weather conditions, air quality, holiday activities, peak hours and other factors have a significant impact on urban traffic conditions. Lin et al. combined diverse new data sources with conventional data, devising a probabilistic framework encompassing a location decomposition model, traffic topic model and traffic speed Gaussian process model. The framework, termed the theme-enhanced Gaussian process aggregation model (TEGPAM), was proposed [25]. Essien et al. built an LSTM model by merging urban road network traffic and weather datasets from Greater Manchester, UK. Experimental results highlighted that temperature variations significantly influenced traffic conditions, indicating its importance as a variable within the model [26]. Li et al. (2019) extracted representative features from the data for fusion, established a prediction model to capture correlation, and used heterogeneous data to predict space-average-velocity [27]. While numerous deep learning methods exist for traffic flow prediction, current research predominantly focuses on univariate traffic time data. There remains a limited extent of research on comprehensive traffic prediction models that integrate multi-source characteristic data fusion.

To sum up, most studies involve parametric and nonparametric prediction models, while some hybrid models are also presented. The deep learning model appears as a “black box” in the learning process and often lacks optimisation for internal parameter design. In addition, in previous studies, the characteristics of traffic speed considered for influence were relatively singular, which could not provide good prediction

accuracy. Therefore, based on the actual data sets of Guangzhou traffic speed and weather conditions, this paper comprehensively considers the spatio-temporal factors that may affect the urban traffic speed, and designs an urban road short-time traffic speed prediction method based on the improved PSO optimised LSTM model. The results may provide potential insights for intelligent transportation systems in traffic state prediction.

3. METHODOLOGY

3.1 LSTM

The LSTM, a specialised type of RNN, adeptly captures long-term dependencies without encountering the gradient vanishing issue. It incorporates a memory cell structure within the hidden layer neurons of RNN to retain past information. Additionally, it integrates three gate structures (input gate, forgetting gate and output gate) to regulate the utilisation of historical data.

If the input sequence is x_1, x_2, \dots, x_T , and the state of the hidden layer is a_1, a_2, \dots, a_Q then at the time q :

$$i_q = \sigma(\omega_{\alpha i} \alpha_{q-1} + \omega_{xi} x_q + b_i) \quad (1)$$

$$\lambda_q = \sigma(\omega_{\alpha \lambda} \alpha_{q-1} + \omega_{\alpha \lambda} x_q + b_\lambda) \quad (2)$$

$$\delta_q = \lambda_q \cdot \delta_{q-1} + i_q \cdot g(\omega_{\alpha \delta} \alpha_{q-1} + \omega_{x\delta} x_q + b_\delta) \quad (3)$$

$$\vartheta_q = \sigma(\omega_{\alpha \vartheta} \alpha_{q-1} + \omega_{x\vartheta} x_q + \omega_{c\vartheta} \vartheta_q + b_\vartheta) \quad (4)$$

$$\alpha_q = \vartheta_q \cdot g(\delta_q) \quad (5)$$

where i_q , λ_q and ϑ_q are the input gate, the forget gate and the output gate respectively; δ_q is the cell unit; ω_α is the weight of the recursive connection; ω_x is the weight from the input layer to the hidden layer; b_i , b_λ , b_δ and b_ϑ are the thresholds of each function; $\sigma(\cdot)$ and $g(\cdot)$ are *sigmoid* and *tanh* functions respectively; \cdot is the dot product.

To align LSTM with the prediction objective, an additional layer of linear regression needs incorporation, namely:

$$\xi_q = \omega_{\xi \vartheta} \alpha_q + b_\xi \quad (6)$$

where ξ_q represents the output of the final prediction result; b_ξ represents the threshold of linear regression.

3.2 Principle of standard PSO

If the potential solution to the optimisation problem is likened to a particle, it continuously navigates through space, adapting its position based on its individual experience and the optimal individual experience while seeking the optimal position. Initially, PSO initialises by acquiring a set of random solutions, then iteratively hunts for the optimal solution by monitoring the current spatially optimal particle. In a multi-dimensional search area, the ψ particles make up a population. In iteration β , the spatial location and speed of the particle α are $\tau_{\alpha,\beta}$ and $\phi_{\alpha,\beta}$ respectively, and the particle updates its position and velocity by supervising the two optimal solutions. The first solution pertains to the optimal solution sought independently by the particle itself, known as the individual extreme value η_α . The second solution refers to the current optimal solution sought by the entire population, denoted as the global optimal solution κ_α . While seeking these two optimal solutions, the particle adjusts its velocity and position utilising the following methods:

$$\phi_{\alpha,\beta+1} = \varepsilon \cdot \phi_{\alpha,\beta} + \chi_1 \cdot v \cdot (\eta_\alpha - \tau_{\alpha,\beta}) + \chi_2 \cdot v \cdot (\kappa_\alpha - \tau_{\alpha,\beta}) \quad (7)$$

$$\tau_{\alpha,\beta+1} = \tau_{\alpha,\beta} + \iota \cdot \phi_{\alpha,\beta+1} \quad (8)$$

where ε is inertial weight; χ_1 and χ_2 are learning factors; v is the random number between $[0,1]$; ι is the velocity coefficient, $\iota=1$.

3.3 Principle of improved PSO

This paper enhances the inertia weight within the PSO algorithm and integrates the mutation mechanism

of the genetic algorithm. This integration bolsters the search capabilities of the PSO algorithm, rendering it adaptable in its exploration abilities.

This algorithm's fundamental concept involves formulating a mathematical model for the inertia weight, allowing for dynamic and adaptive adjustments throughout the entire iterative process. Early in the iteration, the inertia weight quickly diminishes, hastening the algorithm's convergence and refining its local optimisation capacity. Concurrently, the genetic algorithm's mutation mechanism intertwines with regenerating particle positions during iterations to amplify population diversity. This amalgamation enables the algorithm to explore beyond local optima, expanding its search across the solution space and fortifying global optimisation capabilities. The adaptive mutation concept balances the algorithm's local and global search abilities.

In the conventional PSO algorithm, the inertia weight serves to uphold particle motion's momentum while balancing the algorithm's global and local search capabilities. A significant inertia weight strengthens global search, weakens local search, accelerates algorithmic convergence, and yet compromises optimisation accuracy. Conversely, a smaller inertia weight diminishes global search strength, enhances local search, slows the algorithm's convergence, improves search precision, and yet elevates the risk of local optima. The evident conclusion is that the judicious adjustment of the inertia weight value profoundly impacts the algorithm's performance. Setting w will diminish global optimisation capabilities and decelerate the algorithm's convergence rate. Adaptive parameter adjustment is a technical method to improve convergence speed and optimise search performance. This paper suggests a computational approach to enhance PSO performance through the utilisation of nonlinearly changing inertia weights. The calculation *Equation 9* is as follows:

$$\zeta = \zeta_{max} - \frac{\zeta_{max} - \zeta_{min}}{1 + e^{-\frac{\rho}{\rho_{max}}}} \quad (9)$$

where ζ_{max} and ζ_{min} are the upper and lower limits of ζ respectively; ρ is the current iteration number; ρ_{max} is the maximum number of iterations. When ρ is small, ζ approximates ζ_{max} . With the increase of ρ , ζ decreases in a non-linear fashion, while ζ decreases rapidly. This ensures the algorithm's local optimisation capacity while enabling flexible adjustments between its global and local optimisation capabilities.

Inspired by the mutation mechanism in the genetic algorithm, an adaptive variation function is proposed to give particles the ability to jump out of the local range. When $rand \geq prob$, the particles mutate, the particles distribute evenly across the entire space, increasing the randomness of the particles. The *Equations 10–11* for adaptive mutation probability are as follows:

$$\varsigma > \frac{1}{2} \left(1 + \frac{\rho}{\rho_{max}} \right) \quad (10)$$

$$v \geq \varsigma \quad (11)$$

where ς represents the adaptive variogram, which ranges within [0.5, 1]. When ρ is small, ς has a low value, leading to a higher probability of $v \geq \varsigma$, the particles are prone to mutation. It facilitates the particle's escape from local scopes, aiding in global exploration. As ρ increases, as ς goes up, the likelihood of $v \geq \varsigma$ going down increases. Particle mutation probability decreases, lowering the chance of particles getting stuck in local optima. This allows particles to conduct a broad global search before fine-tuning, ensuring comprehensive convergence.

To assess the impact of the enhanced particle swarm optimisation algorithm, the 5-dimensional Sphere test function was individually optimised using the standard PSO algorithm and the IPSO algorithm, followed by a comparative experiment. The two algorithms were iterated 1000 times each. The results were amplified by logarithmic processing, and the fitness curve was obtained as shown in *Figure 1*.

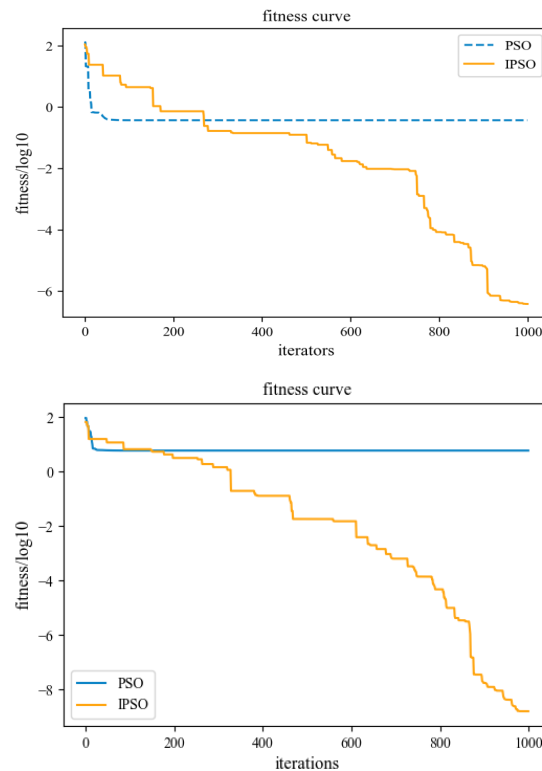


Figure 1 – Fitness curve of Sphere function

Seen from the curve trend after taking logarithms in *Figure 1*, the traditional PSO algorithm falls into the local optimal solution very early in iteration. It no longer has the ability to search for optimisation. The enhanced IPSO algorithm effectively breaks away from local optima, persistently seeking optimal solutions even after 1000 iterations, highlighting its distinct advantages.

3.4 IPSO optimisation of the LSTM algorithm

Leveraging traffic speed and spatio-temporal data, a combined digital-analogue approach is employed for traffic speed prediction. Building upon the aforementioned theory, this study introduces an IPSO-optimised algorithm for the LSTM model.

LSTM models can succumb to overfitting when trained on limited data, thereby jeopardising prediction accuracy. To address this concern, regularisation methods are employed, notably the inclusion of a Dropout layer. This layer selectively deactivates neurons during training, preventing the network from relying too heavily on specific nodes and features. By injecting variability into hidden layer neurons across training iterations, the model's dependency on fixed node relationships is diminished, thus mitigating the risk of feature reliance. Dropout regularisation serves to diminish the impact of individual weight contributions, thereby mitigating model bias and curtailing the risk of overfitting in intricate neural network architectures.

The traffic speed prediction model's framework, integrating diverse data sources, is depicted in *Figure 2*. The algorithmic steps are outlined as follows:

Step 1: Data pre-processing. By analysing the multi-source correlation of velocity data, multiple features with strong correlation were screened out, and the feature sequence of influencing factors was directly used as additional features. The time sequence data was processed in the way of the sliding window, with the window size set to 6, sliding forward one data point at a time, where six time points represent a duration of 1 hour. The model utilises six time-related features to predict the subsequent traffic speed.

Step 2: Initialise the parameters by setting the particle positions and velocities. A population particle $X_{i,0}(a_1, a_2, \partial, l)$, is randomly generated, where a_1 denotes the quantity of neurons in the initial hidden layer;

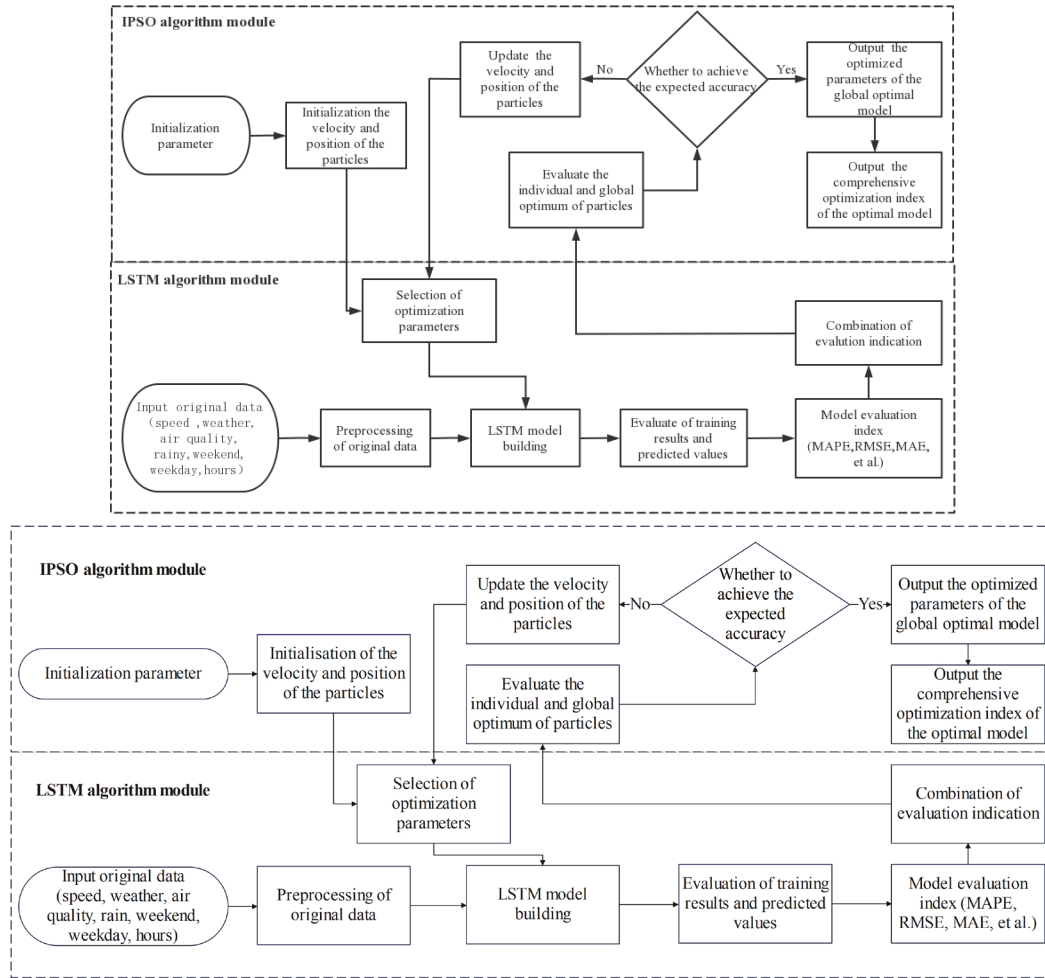


Figure 2 – Flow chart of IPSO optimisation LSTM algorithm

a_2 signifies the quantity of neurons in the secondary hidden layer; ∂ stands for the learning rate within the LSTM; and ι signifies the count of iterations within the LSTM. The population size, iteration count, learning factor, location and speed are established. Each particle is encoded as an array containing four elements, representing the learning rate, the number of iterations, the quantity of neurons in the first hidden layer and the quantity of neurons in the second hidden layer, respectively. By adjusting the ranges of these parameters, we can control the magnitude of parameter changes during mutation, thereby influencing the extent of exploration in the search space. Furthermore, we introduce adaptive mutation operations to avoid premature convergence to local optima. Additionally, by adjusting the range of inertia weights, we influence the magnitude of particle velocity updates during mutation, thereby further optimising the search process of the algorithm, leading to improvements in convergence speed and search efficiency.

Step 3: Establish the particle’s evaluation function by allocating the LSTM parameter to particle $X_{i,0}$ acquired in Step 2. Segment the data into training, validation and prediction samples. The training data are utilised to train the neural network, resulting in the output values \hat{y}_i^j for the training samples and output values \hat{y}_v^k for the validation samples once the iteration limit is reached. The fitness value fit_i for individual X_i is defined using the mean square error (MSE) function as outlined in Equation 12:

$$fit_i = \frac{1}{2} \sum_{j=1}^J (y_i^j - \hat{y}_i^j)^2 + \frac{1}{2} \sum_{k=1}^K (y_v^k - \hat{y}_v^k)^2 \tag{12}$$

where y_i^j and y_v^k represent the anticipated output values of the training and validation samples, respectively. The conventional approach in prior research primarily focuses on the fitting error of training samples as the sole fitness value. However, this practice might lead to suboptimal predictive outcomes if the neural network

becomes overly tailored to the training data, causing overfitting issues. Directly assessing the prediction effectiveness of the model, the validation sample error becomes crucial. Therefore, the fitness function encompasses both the training sample fitting error and the validation sample error. Each carries equal weight, set at 0.5. The model's fitness function is computed by multiplying the sum of these weighted values.

Step 4: Compute the fitness value linked to position X_i for each particle. Individual and population extreme values are established based on the initial fitness value of the particles, with each particle adopting its historical best position as its optimal position.

Step 5: During each iteration, the particle's velocity and position are updated using *Equations 7 and 8*, incorporating both the individual and global extreme values. Following this update, the new fitness value for the particle is computed. The individual and population extreme values are then adjusted based on the fitness values of the updated particles in the population.

Step 6: Once the PSO algorithm reaches its maximum iteration count, the forecasted data obtained are fed into the LSTM model trained using the optimal particle. Consequently, the output comprises the predicted traffic speed.

1.1 Model evaluation indicator

To assess the model's performance, the accuracy of the traffic speed prediction model is appraised using various evaluation metrics. These metrics encompass the mean absolute percentage error (MAPE), root mean square error (RMSE) and mean absolute error (MAE). The specific formulas are denoted in *Equations 13–15*.

$$MAPE = \frac{1}{A} \sum_{r=1}^A \left| \frac{\psi_r^* - \psi_r}{\psi_r} \right| \times 100\% \quad (13)$$

$$RMSE = \sqrt{\frac{1}{A} \sum_{r=1}^A (\psi_r^* - \psi_r)^2} \quad (14)$$

$$MAE = \frac{1}{A} \sum_{r=1}^A |\psi_r^* - \psi_r| \quad (15)$$

In this context, ψ_r^* denotes the forecasted traffic speed yielded by the time series prediction model in period r ; ψ_r stands for the actual traffic speed data, and A signifies the predicted value. Larger MAPE, RMSE and MAE values indicate a greater margin of error within the model.

4. RESULTS

4.1 Data sources

The empirical study utilises traffic speed data provided by the Guangzhou Transportation Commission of China as the experimental dataset. This dataset covers 214 sections, primarily urban expressways and trunk roads, spanning from 1 August 2016 to 30 September 2016, encompassing a 61-day period. Speed data for each segment is aggregated daily within a 10-minute timeframe.

To comprehensively consider the influence of relevant characteristic variables on traffic speed, the characteristic variables affecting urban traffic state are input into the prediction model. Among them, air quality variables include the concentrations of SO₂, NO₂, CO and PM_{2.5} recorded hourly by monitoring stations. Meteorological features include real-time temperature, precipitation and weather conditions recorded by weather stations hour by hour (The codes “0–7” correspond to weather conditions as follows: “sunny, cloudy, cloudy, light rain, moderate rain, heavy rain, shower, rainstorm”). The time attribute is the number of hours (indicating the number of hours in a day within 24 hours) and the duration (indicating the number of days in a week (the code “1” indicates Monday) at the time point. Whether it is a working day (code “1” means a working day and code “0” indicates a non-working day) is also indicated.

4.2 Data processing

We processed and analysed traffic data, revealing that the current traffic speed is similar to recent periods, with the most significant impact from recent times and a decreasing correlation over time. The traffic speed

data exhibit distinct daily and weekly periodic characteristics. There is spatial correlation among traffic speeds across different road segments. We selected external factors, such as PM2.5, SO₂, CO, weather and rainfall, with correlation coefficients exceeding 0.1 with speed data. Subsequently, we normalised various datasets based on these findings. Normalisation of traffic speed and meteorological characteristic data involves a linear transformation of the original data, ensuring that the resultant values are mapped within the [0,1] range. The formula is shown in Equation 16.

$$\mu = \frac{\theta - \theta_{min}}{\theta_{max} - \theta_{min}} \quad (16)$$

where μ is the scaled data; θ is the raw data of traffic speed and meteorological characteristics. The original data ranges from a minimum of θ_{min} to a maximum of θ_{max} . This paper divides the dataset, allocating 90% for training purposes and reserving 10% for testing. An LSTM model optimised through IPSO is utilised to forecast the future traffic speed specifically for Road_1. Sequence data were arranged and reconstructed before and after to construct the feature matrix, as shown in Table 1.

Table 1 – Characteristic matrix of multi-source

Time	Road_1	Road_2	...	Road_40	CO	Weather	PM2.5	...	Rainy	...
2016/8/1 00:10	0.80640	0.91294	...	0.83300	0.83333	1.00000	0.56451	...	0.33246	...
2016/8/1 00:20	0.72673	0.80322	...	0.72879	0.83333	1.0000	0.56451	...	0.33246	...
...
2016/8/31 24:00	0.51809	0.29897	...	0.47534	1.00000	0.20000	0.91936	...	0.23097	...

4.3 Experimental environment

The experiment is based on a python 2.8.1 environment and coded by the TensorFlow1.14.0 deep learning framework. The editing platform is Jupyter Notebook 1.0.0 and the operating system is Windows 10. The study constructed eight prediction models, namely: IPSO-optimised LSTM model with one hidden layer (IPSO-LSTM_1); PSO-optimised LSTM model with one hidden layer (PSO-LSTM_1); IPSO-optimised LSTM model with two hidden layers (IPSO-LSTM2); PSO-optimised LSTM model with two hidden layers (PSO-LSTM_2); General LSTM model; Bidirectional LSTM model (BiLSTM); GRU model; and BP neural network model (BPNN). To ensure experimental fairness, these models employ identical traffic data feature inputs and utilise the uniform Adam optimiser for comparative analysis in traffic speed prediction.

4.4 Initial parameter settings

Table 2 shows the detailed initialisation parameter settings for the IPSO-LSTM model. Details of the LSTM network training and IPSO algorithm optimisation are described below.

Table 2 – Initialise the IPSO-LSTM model hyperparameters

Network parameters	Initial values
Initialisation weight	[0.6,0.8]
Number of training	1000
Population size	5
Individual learning factor	1.5
Population learning factor	0.5
Learning rate	[0.001,0.1]
Neural network capacity	[1,100]
Iteration count	[10,100]
Hidden layer number	2
Batch size	100
Activation function	tanh
Training subset: Validation subset	9:1

4.5 IPSO optimisation parameters determination

Figure 4 illustrates the fitness evolution of the LSTM optimised using particle swarm optimisation. Here, 1 and 2 denote the LSTM model with single and dual hidden layers, respectively. The left one in Figure 4 shows the fitness comparison of IPSO-LSTM_1 and PSO-LSTM_1, while the right one in Figure 4 gives the fitness comparison between IPSO-LSTM_2 and PSO-LSTM_2. The IPSO optimisation of the LSTM demonstrates a reduced fitness value and rapid convergence towards the minimum fitness level. This highlights the IPSO’s superior global optimisation capability and faster convergence rate.

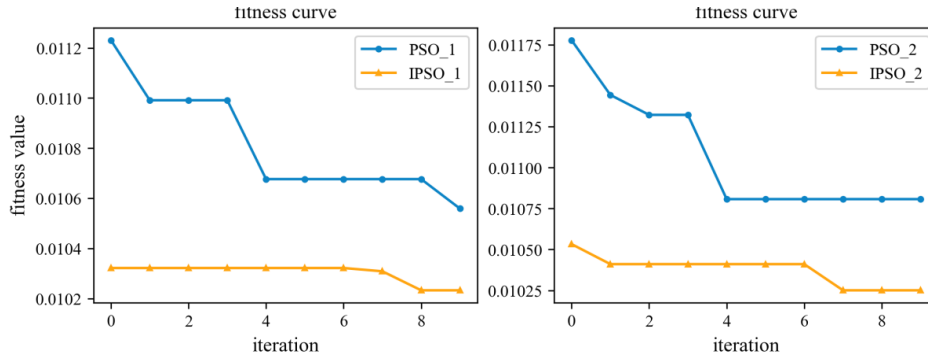


Figure 4 – Comparison of fitness curves

Figure 5 displays the changes in the learning rate, iteration times and hidden layer node number with the iteration times of the IPSO algorithm. Figure 5a indicates the learning rate ϵ fluctuates during iterations before stabilising at 0.0074, while Figure 5b indicates that the iteration number n of the LSTM fluctuates over PSO iterations and eventually steadies at 47 instances. In Figure 5c, the neuron count within the initial hidden layer, h_1 , finally settles at 81. In Figure 5d, the neuron count h_2 in the second hidden layer settles at 12.

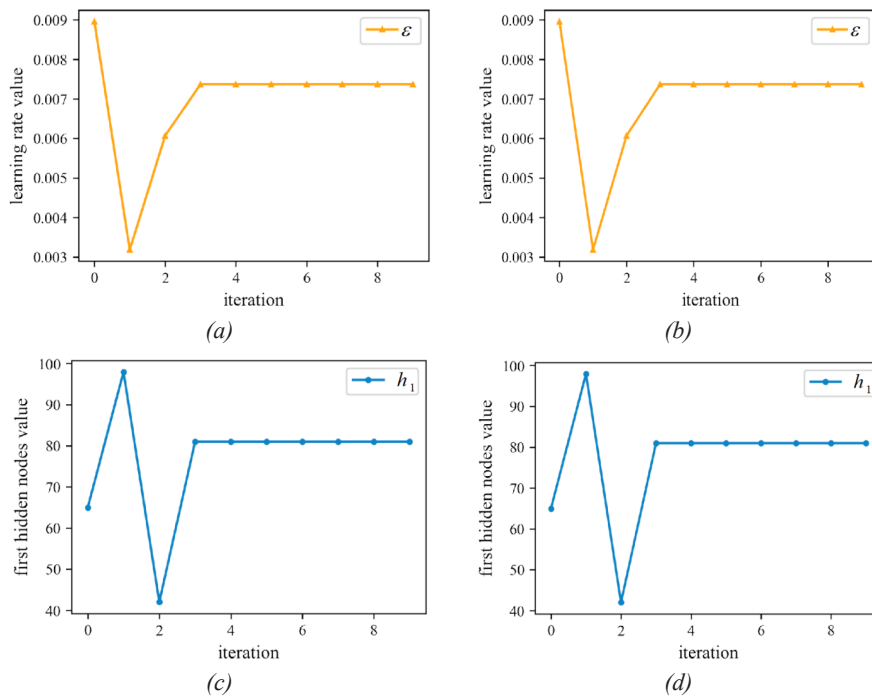


Figure 5 – The optimal values of ϵ , n , h_1 and h_2 in the IPSO-LSTM_2: a) Learning rate; b) Number of iterations of LSTM; c) Number of the neurons within the initial hidden layer; d) Number of the neurons within the secondary hidden layer

5. DISCUSSION

5.1 Effectiveness analysis

The prediction test confirms that IPSO-LSTM_2 outperforms in forecasting traffic speed, as illustrated by the obtained prediction results in Figure 6.

To better explain the prediction performance of the models, *Table 3* compares the prediction performance of each model using identical sample inputs. It can be seen that in comparison to the MAPE of BPNN, LSTM, GRU, Bi-LSTM, Conv-LSTM, CNN-LSTM and the prediction accuracy of IPSO-LSTM_2 is improved by 20.96%, 14.24%, 26.13%, 23.36%, 22.73% and 13.90%, respectively, showing the best prediction performance. Therefore, following the enhancement of the particle swarm optimisation algorithm and its integration with the LSTM, the model fitting effect is effectively enhanced and the prediction accuracy is significantly improved.

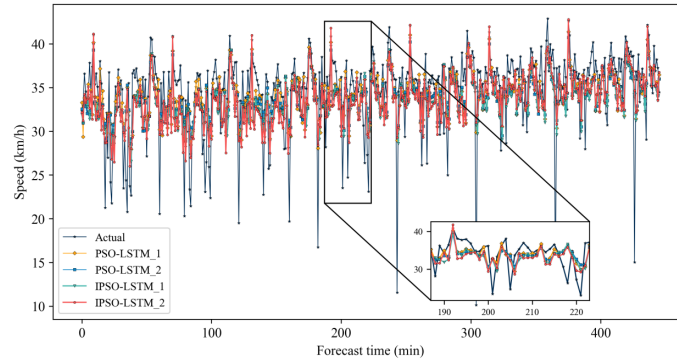


Figure 6 – Prediction results of LSTM optimised by PSO and IPSO

Table 3 – Prediction errors of other models

Errors	BPNN	LSTM	GRU	Bi-LSTM	Conv-LSTM	CNN-LSTM	IPSO-LSTM_2
MAPE (%)	0.12006	0.11065	0.12846	0.12381	0.12280	0.11021	0.09489
RMSE	4.71527	4.40562	4.90083	5.21896	4.56724	4.36695	3.93772
MAE	3.50607	3.12706	3.21892	3.32392	3.18591	3.06029	2.62404

To validate the effectiveness of our model across different prediction time ranges, we conducted prediction experiments on various time intervals using the Guangzhou traffic speed dataset. The specific results are presented in *Table 4*.

Table 4 – Comparison of prediction performance at different time intervals

Errors	MAPE (%)	RMSE	MAE
10 min	0.09489	3.93772	2.62404
20 min	0.07392	3.16551	2.33185
30 min	0.06265	2.62200	2.04044

We found that our model performs excellently in the prediction tasks of 10, 20 and 30 minutes, with performance improving as the prediction time extends.

5.2 Robustness analysis

After conducting experiments on the dataset from Guangzhou, we have demonstrated the robustness and generalisation capability of our model. To further validate our findings, we compared it with similar traffic speed data from Shenzhen, as shown in *Table 5*.

Table 5 – Comparison of predictive performance of different datasets

Errors	MAPE (%)		RMSE		MAE	
	PSO-LSTM_2	IPSO-LSTM_2	PSO-LSTM_2	IPSO-LSTM_2	PSO-LSTM_2	IPSO-LSTM_2
Guangzhou	0.11055	0.09489	4.26611	3.93772	3.37811	2.62404
Shenzhen	0.11154	0.10660	5.84341	5.11804	4.72913	3.92970

The experimental results indicate that our model IPSO-LSTM_2 outperforms PSO-LSTM_2 in traffic speed prediction tasks in both Guangzhou and Shenzhen, confirming the robustness and generalisation of our model. We conducted an analysis of traffic speed on weekdays and weekends, revealing significant daily and

weekly periodic features in the data. On weekdays, the traffic speed time series for the same road segment is similar to the previous day. Although the patterns differ between weekends and weekdays, the similarity between weekends also serves as an important basis for traffic prediction. Additionally, we used the IPSO-LSTM_2 model for prediction, and the accuracy curves are shown in *Figure 7c*. It can be observed from the figure that our model demonstrates good performance and accuracy in both weekday and weekend speed prediction tasks.

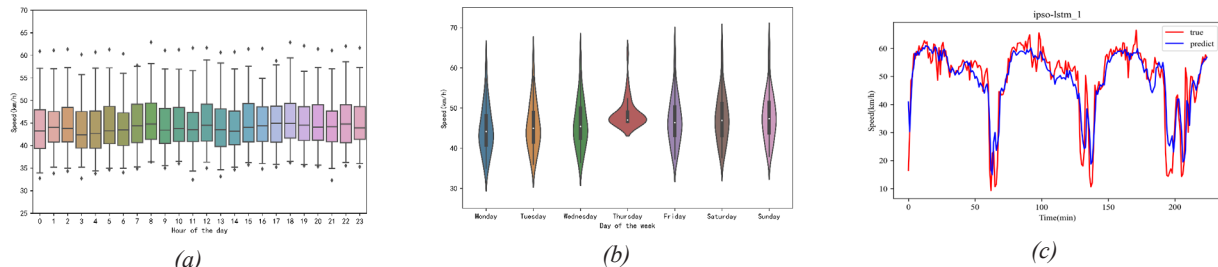


Figure 7 – Periodic characteristics and prediction accuracy analysis: a) Speed data of all sections in a day; b) Speed data of all sections in a week; c) The prediction accuracy curve

6. CONCLUSIONS

This study proposes a traffic speed prediction model based on multi-feature data fusion and IPSO-LSTM. Firstly, we optimised the PSO by integrating nonlinear weight variations and incorporating a mutation mechanism with genetic algorithms. These modifications enhanced the global optimisation capability of the PSO and accelerated its convergence speed. The fusion of the PSO and the LSTM involves optimising the number of hidden neurons, learning rate and iterations of the LSTM using the PSO. This approach alleviates the limitations of manually defining LSTM parameters, thereby improving the accuracy of traffic speed prediction. Compared to the PSO, the IPSO achieves superior optimisation of the LSTM parameters, enhancing the overall predictive performance. Furthermore, we input data related to speed prediction, such as PM2.5, SO₂, CO, weather and rainfall, into the model to improve the accuracy of traffic speed prediction. Lastly, evaluation of performance metrics such as the MAPE, RMSE and MAE reveals that the proposed LSTM neural network, integrated with multi-source data fusion and IPSO optimisation, outperforms other baseline models in terms of prediction accuracy and demonstrates good performance across different datasets and time intervals.

In summary, the proposed speed prediction method shows promising feasibility and provides a tailored research foundation for urban road traffic management. Considering the limited availability of device resources, this study conducted model training with a small sample size. Future exploration could focus on further researching big data fusion techniques, expanding the experimental sample, and more extensive applications in traffic state monitoring research.

ACKNOWLEDGEMENTS

This research was supported by the Natural Science Foundation of China (No.52062027), the Industry Support Plan Project from Department of Education of Gansu Province(No.2024CYZC-28), the “Double-First Class” Major Research Programs, Educational Department of Gansu Province (No.GSSYLXM-04), the Key Research and Development Project of the Gansu Province (No.22YF7GA142), the Soft Science Special Project of the Gansu Basic Research Plan under Grant No.22JR4ZA035, the Gansu Provincial Science and Technology Major Special Project – the Enterprise Innovation Consortium Project (No.22ZD6GA010), the Natural Science Foundation of the Gansu Province (22JR5RA343), the Foundation of A Hundred Youth Talents Training Program of the Lanzhou Jiaotong University, China, and the Open Fund of the National Engineering Research Centre of Highway Maintenance Technology, Changsha University of Science & Technology (No.kfj220108).

REFERENCES

- [1] Wang J, Chen R, He Z. Traffic speed prediction for urban transportation network: A path based deep learning approach. *Transportation Research Part C: Emerging Technologies*. 2019;100:372-385. DOI: 10.1016/j.trc.2019.02.002.

- [2] Chandra SR, Al-Deek H. Predictions of freeway traffic speeds and volumes using vector autoregressive models. *Journal of Intelligent Transportation Systems*. 2009;13(2):53-72. DOI: 10.1080/15472450902858368.
- [3] Sun J, Sun J. A dynamic Bayesian network model for real-time crash predictions using traffic speed conditions data. *Transportation Research Part C: Emerging Technologies*. 2015;54:176-186. DOI: 10.1016/j.trc.2015.03.006.
- [4] Wang H, et al. A novel work zone short-term vehicle-type specific traffic speed prediction model through the hybrid EMD-ARIMA framework. *Transportmetrica B-Transport Dynamics*. 2016;4(3):159-186. DOI: 10.1080/21680566.2015.1060582.
- [5] Zhang W, et al. Speed prediction based on a traffic factor state network model. *IEEE Transactions on Intelligent Transportation Systems*. 2021;22(5):3112-3122. DOI: 10.1109/TITS.2020.2979924.
- [6] Zeng C, Ma C, Wang K, Cui Z. Predicting vacant parking space availability: A DWT-Bi-LSTM model. *Physica A: Statistical Mechanics and its Applications*. 2022;599:127498. DOI: 10.1016/j.physa.2022.127498.
- [7] Ma C, Zhao M. Spatio-temporal multi-graph convolutional network based on wavelet analysis for vehicle speed prediction. *Physica A: Statistical Mechanics and its Applications*. 2023;630:129355. DOI: 10.1016/j.physa.2023.129233.
- [8] Asif MT, et al. Spatiotemporal patterns in large-scale traffic speed prediction, *IEEE Transactions on Intelligent Transportation Systems*. 2014;15(2):794-804. DOI: 10.1109/TITS.2013.2290285.
- [9] Yao B, et al. Short-term traffic speed prediction for an urban corridor. *Computer-aided Civil and Infrastructure Engineering*. 2017;32(2):154-169. DOI: 10.1111/mice.12221.
- [10] Tang J, et al. An improved fuzzy neural network for traffic speed prediction considering periodic characteristic. *IEEE Transactions on Intelligent Transportation Systems*. 2017;18(9):2340-2350. DOI: 10.1109/TITS.2016.2643005.
- [11] Zang D, et al. Long-term traffic speed prediction based on multiscale spatio-temporal feature learning network. *IEEE Transactions on Intelligent Transportation Systems*. 2019; 20(10):3700-3709. DOI: 10.1109/TITS.2018.2878068.
- [12] Zheng L, Huang H, Zhu C, Zhang K. A tensor-based K-nearest neighbors method for traffic speed prediction under data missing. *Transportmetrica B-Transport Dynamics*. 2020;8(1):182-199. DOI: 10.1080/21680566.2020.1732247.
- [13] Zhang K, Zheng L, Liu Z, Jia N. A deep learning based multitask model for network-wide traffic speed prediction. *Neurocomputing*. 2020;396:438-450. DOI: 10.1016/j.neucom.2018.10.097.
- [14] Zhang Z, et al. Multistep speed prediction on traffic networks: A deep learning approach considering spatio-temporal dependencies. *Transportation Research Part C: Emerging Technologies*. 2019;105:297-322. DOI: 10.1016/j.trc.2019.05.039.
- [15] Zheng G, Chai W, Katos, V. A dynamic spatial-temporal deep learning framework for traffic speed prediction on large-scale road networks. *Expert Systems with Applications*. 2022;195:116585. DOI: 10.1016/j.eswa.2022.116585.
- [16] Ma C, et al. A novel STFSA-CNN-GRU hybrid model for short-term traffic speed prediction. *IEEE Transactions on Intelligent Transportation Systems*. 2023;24(4):3728-3737. DOI: 10.1109/TITS.2021.3117835.
- [17] Ma C., Zhao M. Optimized deep extreme learning machine for traffic prediction and autonomous vehicle lane change decision-making. *Physica A: Statistical Mechanics and its Applications*. 2023;633:129355. DOI: 10.1016/j.physa.2023.129355.
- [18] Ma X, et al. Long short-term memory network for traffic speed prediction using remote microwave sensor data. *Transportation Research Part C: Emerging Technologies*. 2015; 54:187-197. DOI: 10.1016/j.trc.2015.03.014.
- [19] Jia Y, et al. Rainfall-integrated traffic speed prediction using deep learning method. *IET Intelligent Transport Systems*. 2017;11(9):531-536. DOI: 10.1049/iet-its.2016.0257.
- [20] Gu Y, et al. Short-term prediction of lane-level traffic speeds: A fusion deep learning model. *Transportation Research Part C: Emerging Technologies*. 2019; 106:1-16. DOI: 10.1016/j.trc.2019.07.003.
- [21] Hu H, Lin Z, Hu Q, Zhang Y. Attention mechanism with spatial-temporal joint model for traffic flow speed prediction. *IEEE Transactions on Intelligent Transportation Systems*. 2021;23(9):16612-16621. DOI: 10.1109/TITS.2021.3113935.
- [22] Meng X, et al. D-LSTM: Short-term road traffic speed prediction model based on GPS positioning data. *IEEE Transactions on Intelligent Transportation Systems*. 2022;23(3):2021-2030. DOI: 10.1109/TITS.2020.3030546.

- [23] Zeng C, Ma C, Wang K, Cui Z. Parking occupancy prediction method based on multi factors and stacked GRU-LSTM. *IEEE Access*. 2022;10:47361-47370. DOI: 10.1109/ACCESS.2022.3171330.
- [24] Ma C, Dai G, Zhou J. Short-term traffic flow prediction for urban road sections based on time series analysis and LSTM_BILSTM method. *IEEE Transactions on Intelligent Transportation Systems*. 2022;23(6):5615-5624. DOI: 10.1109/TITS.2021.3055258.
- [25] Lin L, et al. Road traffic speed prediction: A probabilistic model fusing multi-source data. *IEEE Transactions on Knowledge and Data Engineering*. 2018;30(7):1310-1323. DOI: 10.1109/TKDE.2017.2718525.
- [26] Essien A, Petrounias I, Sampaio P, Sampaio S. Improving urban traffic speed prediction using data source fusion and deep learning. *IEEE International Conference on Big Data and Smart Computing (BigComp) 2019*, 27 Feb. 2019, Kyoto, Japan. 2019. p. 331-338. DOI: 10.1109/BIGCOMP.2019.8679231.
- [27] Li L, et al. Traffic speed prediction for intelligent transportation system based on a deep feature fusion model. *Journal of Intelligent Transportation Systems*. 2019;23(6):605-616. DOI: 10.1080/15472450.2019.1583965.

赵永鹏, 荔永苍, 马昌喜, 王珂, 徐学才

利用多源数据融合优化的LSTM神经网络进行交通速度预测

摘要:

准确实时地预测交通速度对智能交通系统的发展至关重要。鉴于车辆数据的非线性和随机性, 结合多源时空数据与改进的粒子群优化算法(IPSO)为优化长短期记忆神经网络(LSTM)提供了一种有前景的方法。首先, 我们通过引入非线性惯性权重和自适应变异提升了PSO的优化能力。其次, 为解决LSTM超参数选择的难题, PSO算法通过迭代训练有效地识别全局最优解, 确保合适的参数设置。随后, 我们使用多源时空交通速度数据进行案例研究, 将我们提出的IPSO-LSTM模型与传统神经网络预测模型及先进模型进行了比较。实验结果表明, 本研究提出的IPSO-LSTM模型解决了传统LSTM模型在交通状态预测中遇到的参数选择问题和预测不准确问题。此外, 它增强了模型捕捉速度时间序列动态的能力。值得注意的是, 在处理复杂速度数据时, 我们的模型在预测准确性和稳定性方面表现出显著优势。

关键词:

交通速度预测; 多源数据融合; 改进的粒子群优化; 长短期记忆网络