Available online www.jsaer.com

***Journal of Scientific and Engineering Research*, 2023, 10(11):72-79**

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

PSO-GRU Based Short-Term Traffic Flow Prediction

Yingping Tang[1*], Longjiao Yin[1]

[1]School of Transportation and Vehicle Engineering, Shandong University of Technology, Zibo 255000, China.
*Email: tangyingping0810@163.com

Abstract In order to further improve the accuracy of short-term traffic flow prediction, a short-term traffic flow prediction method based on particle swarm (PSO) optimised gated recurrent unit (GRU) is proposed. And PSO is used to find the optimal parameter combinations of the GRU, and the parameters to be optimised are the initial learning rate of the neural network, the learning descent factor and the number of neurons in the hidden layer. In order to verify the effectiveness of the proposed model, the traffic flow data provided by PEMS database is used for example verification. The results show that the prediction model proposed in this paper is able to describe the law of traffic flow changes well and has higher prediction accuracy than the traditional BP and SVM models.

Keywords traffic flow prediction; particle swarm optimisation; gated recurrent unit

1. Introduction

Traffic flow prediction is an important part of traffic management control and guidance, but also an indispensable part of urban traffic intelligence, which is of great practical significance to improve the level of road service and control. Urban traffic system is a dynamic, stochastic, non-linear complex system, how to improve the accuracy of urban short-term traffic flow prediction is one of the challenges faced by domestic and foreign traffic practitioners at this stage, but also the hot content of the research.

Domestic and foreign scholars mainly use parametric model and non-parametric model to forecast short-term traffic flow: The parametric model mainly uses Markov chain prediction model, Kalman filter prediction model [1], autoregressive comprehensive moving average (ARIMA) model [2], exponential smoothing algorithm [3]and other mathematical models to forecast. However, these models have their limitations when dealing with complex traffic environment. Non-parametric models are mainly used to predict future traffic conditions by analyzing potential laws and connections between historical data, and support vector machine [4], K-nearest neighbor algorithm [5], artificial neural network [6], Bayesian network [7] and other methods are commonly used. With the introduction and popularization of intelligent algorithms, many scholars use them to improve the performance of traffic flow prediction models. In literature [8], hybrid particle swarm optimization was used to train the structure and parameters of neural networks (NN) to increase the prediction speed. Literature [9] uses improved genetic algorithms to set appropriate parameters for neural networks and establish self-regulating neural networks to improve prediction accuracy. Literature [10] used the improved cuckoo search algorithm to optimize the parameters of RBF neural network for traffic flow prediction.

The current short-time traffic flow prediction methods generally have shortcomings such as poor real-time performance, slow optimisation speed and too many model assumptions, which seriously affect the practical application of short-time traffic flow prediction. Aiming at the problems of the existing prediction methods, this paper proposes a short-time traffic flow prediction method based on particle swarm optimisation of gated cyclic units, and uses the traffic flow data provided by the PEMS database to carry out example validation and test the effectiveness of the model.

2. GRU

GRU is an improvement of RNN, through the introduction of the "gate" structure to effectively overcome the problem of gradient disappearance and gradient explosion in RNN, its network neurons mainly contain two "gate" structures, respectively, the update gate and the reset gate, the structure of the gated recurrent unit is shown in Figure 1, where the update gate z_t achieves selective forgetting of the hidden state information of the previous moment, and the reset gate r_t achieves selective memory of the current state and information. The structure of gated recurrent unit is shown in Fig. 1, and the arrow direction indicates the data flow direction, among which, the update gate z_t realises the selective forgetting of the hidden state information in the previous moment, and the reset gate r_t realises the selective remembering of the current state and information, and the closer the value of *zt* and *Rt* is to 0, it means that the more information needs to be forgotten at the previous moment; and the closer the value of z_t and r_t is to 1, it means the more information is remembered, and the calculation formula of GRU is as follows:

$$r_t = \sigma(W_Z \Box [h_{t-1}, x_t]) \quad (1)$$

$$z_t = \sigma(W_r \Box [h_{t-1}, x_t]) \quad (2)$$

$$h_t = \tanh(W_h \Box [r_t * h_{t-1}, x_t]) \quad (3)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * h_t \quad (4)$$

where: r_t is the reset gate; z_t is the update gate; x_t is the current input; h_t is the aggregation of the current input and the previous hidden layer states; h_t is the hidden layer output after updating the memory; W_Z and W_r are the weight matrices of the reset gate and the update gate, respectively; W_h is the weight matrix of the hidden layer; σ and tanh are the Sigmoid activation function and the Hyperbolic Tangent activation function, respectively.

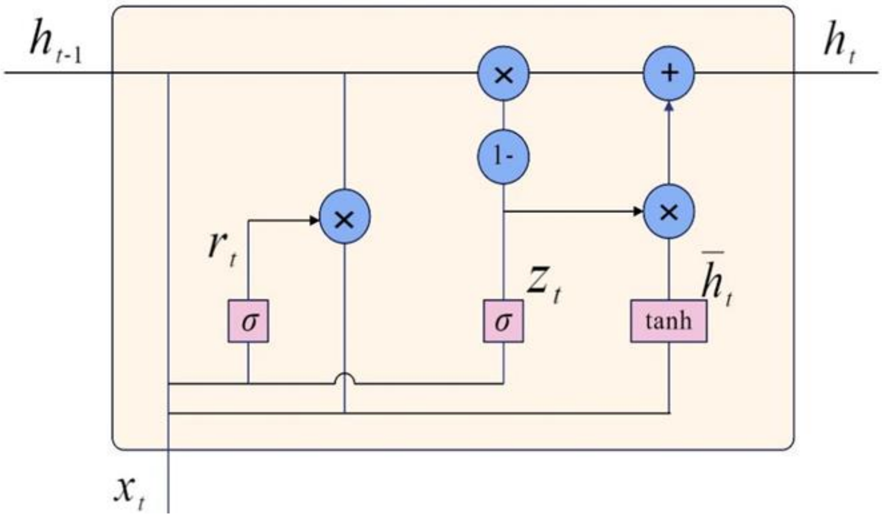

Figure 1: GRU structure

3. PSO-GRU

The emergence of GRU is a good solution to the long-term dependence problem, but during the training process, GRU still has the problems of the network structure is difficult to determine and how to choose the optimal parameters. Based on the above problems, this paper uses PSO to optimise GRU and construct a prediction model based on PSO-GRU. The PSO algorithm is used to optimise the initial learning rate of GRU, the learning descent factor, and the number of neurons in the hidden layer to determine the optimal parameter combination. The specific steps of the model are as follows:

(1) Initialise the particle population parameters. Determine the population size, the maximum number of iterations, the search dimension, the acceleration factor, and the inertia weights.

(2) Initialise the position and velocity of the particle. A particle is randomly generated, $X_i = (k, l, n)$, *k* denotes the initial learning rate of the GRU, *l* denotes the learning descent factor, and *n* denotes the number of neurons in the hidden layer, where the range of values of *k* is set to [6e-3, 4e-3], the range of values of *l* is set to [0.1, 1], and the range of values of n is set to [1,30], and the particle's velocity $V_i = (V_{i1}, V_{i2}, V_{i3})$, the value of the velocity of the particle for each dimension is taken by the value of each dimension variable determination.

(3) Constructing the GRU structure. In this study, the neural network structure is set up with a total of five layers, which are input layer, hidden layer, discard layer, fully connected layer, and regression layer. The number of neurons in the hidden layer is determined by PSO optimisation, the discard layer is to prevent the network from overfitting, and the ratio is generally set to 0.2, and the loss function of the regression layer is the mean square error of the predicted response at each time, and the optimiser is Adam, and the loss function is calculated by the formula:

For a single observation, the mean square error is:

$$MSE = \sum_{i=1}^{R} \frac{(t_i - y_i)^2}{R} \qquad (5)$$

where *R* is the number of responses, t_i is the target output, y_i is the network's prediction for response *i*.

$$loss = \frac{1}{2S} \sum_{i=1}^{S} \sum_{j=1}^{R} (t_{ij} - y_{ij})^2 \qquad (6)$$

where *S* is the sequence length.

(4) Determine the fitness function of the particle swarm optimisation algorithm. The preprocessed training data is input into the input layer, and a population particle $X_i = (k, l, n)$ is randomly generated in step 2 as the initial parameter values of the GRU neural network, and the mean square error (*MSE*) between the measured and predicted values of the validation set is used as the fitness function, which is calculated by the formula:

$$f = \frac{1}{n} \sum_{i=1}^{n} (y_i - y_i)^2 \qquad (7)$$

Where, y_i represents the predicted value, y_i represents the true value, and *n* is the number of samples.

(5) Particle replacement optimization. The particles are constantly compared with the historical optimal positions of the particles in the process of movement, and at the same time, all the particles are compared with the global historical optimal positions, so that the new local optimal positions and global optimal positions can be obtained. When the optimal position obtained by replacement meets the stopping criterion, the parameter combination at this time will be used as the optimal parameters of the GRU model for training and prediction, and vice versa, the particle optimisation will be carried out again.

(6) Model prediction. The optimal parameters determined by the PSO algorithm are used to train the GRU to reach the stopping criterion, and the test set data are input to the trained model for prediction, and the final prediction results are output.

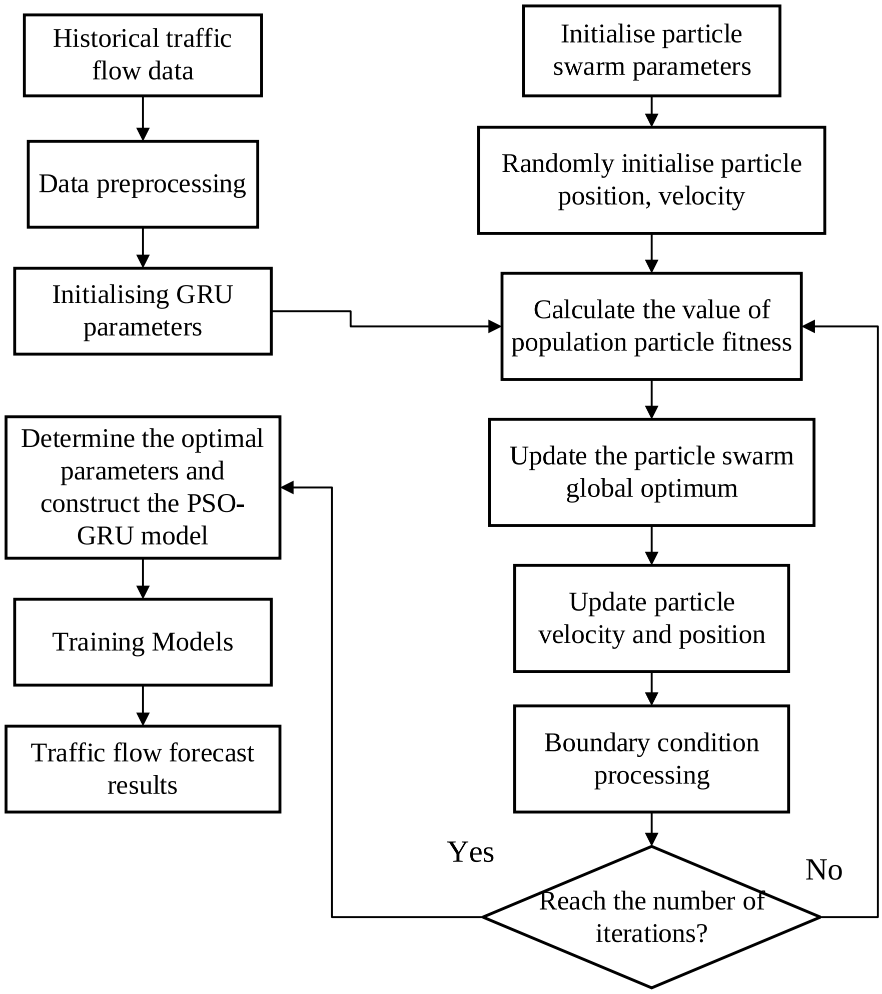

Figure 2: Flow of the PSO-GRU prediction model

4 Experiment

4.1 Data Selection

The traffic flow data used in the example validation of this article is sourced from the open database of the California Transportation Administration's Performance Testing System (PeMS) in the United States. We selected the traffic flow data of detector number 1113028 to validate the performance of our proposed model. The traffic flow data we obtained from the detector was from January 4, 2021 to January 8, 2021. Train with traffic flow data from the first four days and predict traffic flow data from the fifth day. Collect data every 5 minutes, which means 288 data samples can be obtained every day.

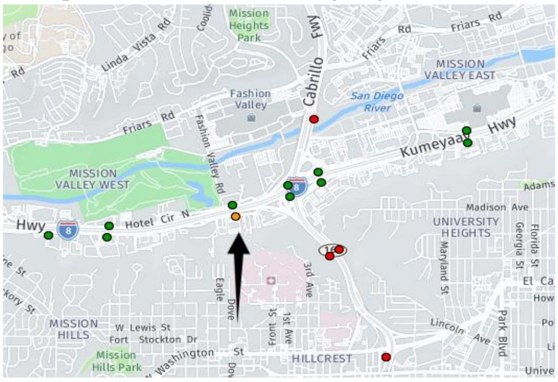

Figure 3: Selected detectors

4.2 Evaluation metrics

In order to evaluate the performance of the prediction method, the evaluation indexes we use are root mean square error (*RMSE*), mean absolute error (*MAE*). The smaller the value of the *RMSE* and *MAE* indexes, the higher the accuracy of the prediction, and their formulas are as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(f_i - f_i)^2} \quad (8)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}(f_i - f_i) \quad (9)$$

Where, f_i denotes real traffic flow data; f_i denotes predicted data; *n* is the number of samples.

4.3 Experimental Environment

The computer configuration and software environment used for model training and testing is the processor is AMD Ryzen 5 6600H with Radeon Graphics and 16.0GB of RAM; the system is Windows 11 (64-bit) ; the programming language version is MATLAB R2022b.

4.4 Experimental Data Preprocessing

In order to improve the convergence speed of the model, the sample data after preprocessing is normalised to within the interval [0,1], and the corresponding mathematical expression is:

$$X = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (10)$$

In the formula: *X* is the normalised data; *x* is the original data; $x_{\min}$ and $x_{\max}$ are the minimum and maximum values of the original data respectively.

4.5 Experimental results and analysis

PSO is used to find the optimal parameter combination of GRU, the parameters that need to find the optimal are the initial learning rate of GRU, the learning descent factor, and the number of neurons in the hidden layer, and the results of the search for the optimal are as follows:

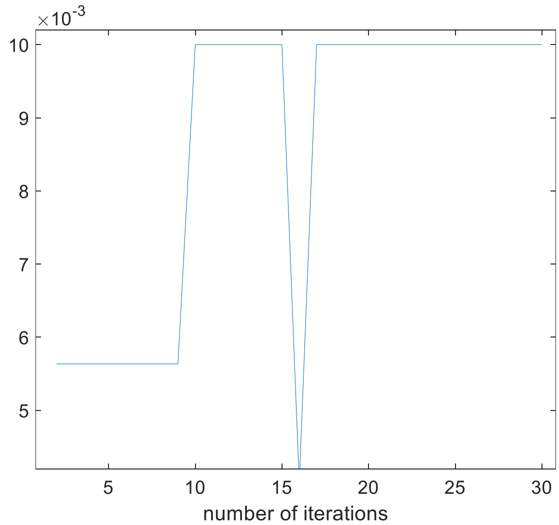

Figure 4: Optimising the initial learning rate

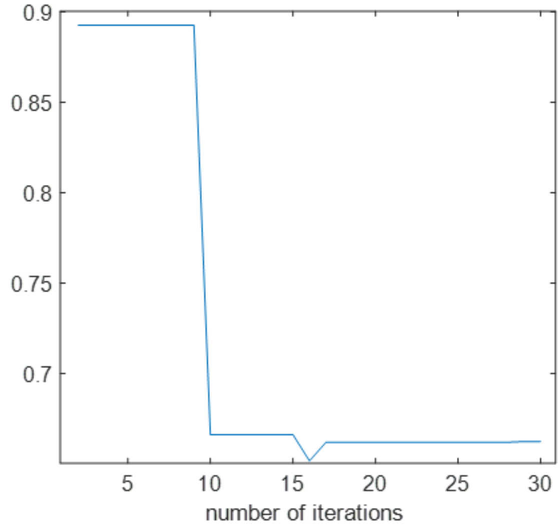

Figure 5: Optimisation of the learning rate degradation factor

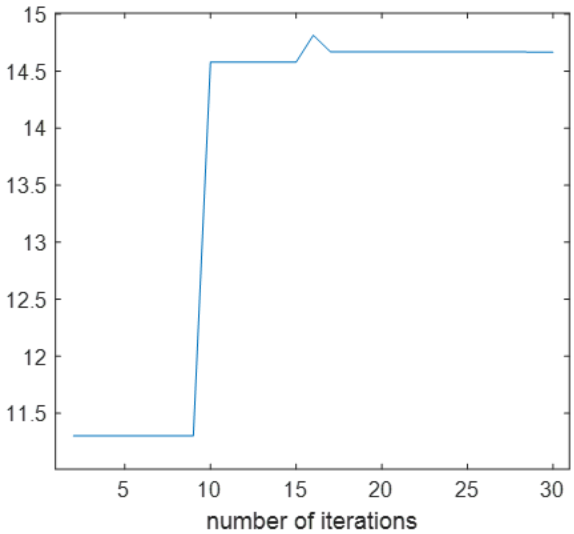

Figure 6: Optimising the number of neurons in the hidden layer

From Figures 4 to 6, the optimal hyperparameters initial learning rate, learning descent factor, and the number of hidden layer neurons for the PSO found GRU model are 0.01, 0.663, and 15, respectively, because the number of hidden layer neurons must be an integer.

After many model training debugging, the final model PSO-GRU parameters training batch size is set to 128, the maximum number of training times is 600, the time step is 10, the Adam optimiser is used, the initial learning rate, the learning descent factor, and the number of neurons in the hidden layer are 0.01, 0.663, and 15, respectively.

In order to verify the effectiveness of our proposed prediction model PSO-GRU, we also used SVM, BP for comparison. Two evaluation metrics, *RMSE* and *MAE*, were used to evaluate the effectiveness of the three model predictions, and the results of traffic flow prediction are as follows:

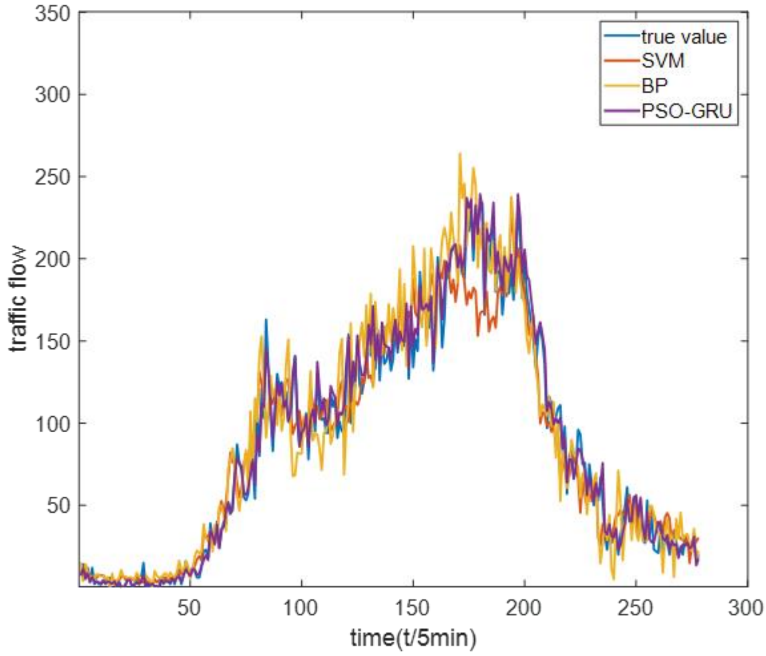

Figure 7: Traffic flow prediction results

From Fig. 7, we can see that SVM, BP and PSO-GRU predict the traffic flow curve of the detector on the fifth day, and it can be found that the traffic flow curve predicted by the proposed model PSO-GRU is the closest to the traffic flow curve of the real value, and it is highly fitted to the traffic flow curve of the real value.

Table 1: Comparison of prediction error results

Predictive modelling	*RMSE*	*MAE*
SVM	13.7536	8.7653
BP	11.6428	8.2147
PSO-GRU	6.3975	5.4186

Table 1 shows the comparison results of the evaluation indexes of the three methods. As can be seen from Table 1: the *RMSE* and *MAE* of short-term traffic flow prediction based on the PSO-GRU model proposed in this paper decreased significantly compared with the other two methods, more specifically, the proposed model decreased 53% in *RMSE* and 38% in *MAE* compared with SVM, and decreased 45% in *RMSE* and 34% in *MAE* compared with BP. In summary, our proposed PSO-GRU model predicts significantly better accuracy and has great advantages.

5 Conclusion

This paper proposes a short-term traffic flow prediction method based on particle swarm optimisation gated recurrent unit (PSO-GRU), which improves the ability of the model to generalise and seek the global optimum, and the results of the example validation show that: the PSO-GRU model is able to accurately predict the short-term traffic flow, and a comparison of the prediction results with those of the BP and SVM models further illustrates the model's superiority in the prediction of short-term traffic flow, and it is able to improve the traffic flow prediction accuracy.

References

[1] Yang Z, Zhu Z. A real-time traffic flow prediction model based on Kalman filter theory [J]. China Highway Journal, 1999, 12(3): 63-67.

[2] Kumar S V, Vanajakshi L. Short-term traffic flow prediction using seasonal ARIMA model with limited input data [J]. European Transport Research Review, 2015, 7(3): 1-9.

[3] Williams B M, Durvasula P K, Brown D E. Urban freeway traffic flow prediction: application of seasonal autoregressive integrated moving average and exponential smoothing models[J]. Transportation Research Record, 1998, 1644(1): 132-141.

[4] Chen Lei, Zhang Tuqiao. A temporal water consumption prediction model based on least squares support vector machine [J]. Journal of Harbin Institute of Technology, 2006, 38(9): 1528-1530.

[5] Bereau M, Dubuisson B. A fuzzy extended k-nearest neighbors rule [J]. Fuzzy sets and systems, 1991, 44(1): 17-32.

[6] Gao D, Wang P, Cai Z. Optimisation of hidden layer nodes and training times in artificial neural networks [J]. Journal of Harbin Institute of Technology, 2003, 35(2): 207-209.

[7] Wang Y, Gao H, Chen X. Traffic flow prediction using variable structure dynamic Bayesian network [J]. Computer Science and Exploration, 2017, 11(4): 528-538.

[8] Liu S, Li D, Xi Y, et al. A short-term traffic flow forecasting method and its applications [J]. Journal of Shanghai Jiaotong University (Science), 2015, 20: 156-163.

[9] Hashemi S M H, Moeinfar R M, Rahimipour S R. Traffic prediction using a self-adjusted evolutionary neural network [J]. 2018.

[10] Cai Y, Le B, Cai R, et al. Short-time traffic flow prediction on highways under heavy rainfall [J]. Computer Engineering, 2020, 46(6): 34-39.