ISSN 1330-3651 (Print), ISSN 1848-6339 (Online) https://doi.org/10.17559/TV-20240522001637
Original scientific paper

# Traffic Congestion Prediction on Urban Roads Based on Multiple Types of External Datasets

Jichen WANG

**Abstract:** With the acceleration of urbanization and the increasing demand for human travel, the problem of traffic congestion on urban roads has become increasingly serious. Strengthening the construction of Intelligent Transportation System (ITS) and improving the accuracy of traffic flow prediction are the core issues to slow down traffic congestion and traffic accidents. This study selects the section of Xizhimen North Avenue in Beijing as the research object, introduces external variables such as subway station passenger flow, holiday information and meteorological data, and adopts four models, namely, SARIMA, SVR, GRU and LSTM, for traffic flow prediction. The results show that the deep learning-based LSTM and GRU models are excellent in prediction performance, with $R^2$ values over 0.79 and MAE and MSE values less than 0.1, which are significantly better than SVR and SARIMA models. These results validate the important influence of external factors on traffic flow and demonstrate the potential of deep learning methods in traffic prediction, which provide a scientific decision-making basis for the traffic management department and help to formulate more effective traffic diversion and management strategies to alleviate the urban traffic pressure.

**Keywords:** deep learning; long short-term memory; traffic flow prediction

## 1 INTRODUCTION

In today's era, with the development of technology and the increasing demand for human mobility, traffic congestion has brought about serious negative impacts on urban economic development and ecological and environmental protection. The Global status report on road safety 2023 released by the World Health Organization reports that the number of road traffic deaths has declined to 1.19 million, but among children or young people between the ages of 5-29, traffic accident casualties are still the number one killer of this group. This also shows that making timely and accurate traffic forecasts and strengthening road safety measures are the common goal of all mankind.

As an important part of intelligent transportation system research [1, 2], traffic flow prediction can accurately estimate the traffic flow in a specific area within a specific time interval in the future, remind the traveling public and the relevant departments, coordinate road traffic route planning in a timely manner, effectively alleviate the traffic pressure, improve the safety and cost-effectiveness of transportation, and optimize the construction of future urban development [3]. Due to the characteristics of traffic data, the traffic flow prediction problem has a complex nonlinear spatio-temporal dependence and is widely dependent on external factors [4], such as weekdays, holidays, weather, road conditions, public events, etc. [5], and the different external factor conditions will have a significant impact on road friction coefficients, the residents' willingness to travel, and the driver's operation, which will bring about a great change in the traffic flow of the urban roads.

As the capital of the People's Republic of China, Beijing is the political and cultural center of the country, a world-famous ancient capital and a modern international city, as well as a gathering place for talents from home and abroad. According to the records of Beijing Municipal Bureau of Statistics, as of the end of 2023, the city's resident population was 21.858 million, an increase of 15,000 from the end of the previous year, which ranked the third in the list of cities in China. Meanwhile, many tourists from home and abroad came to the city, and Beijing received a total of 329 million tourists in 2023, an increase of 80.2% from the previous year, and realized a total tourism revenue of 584.97 billion yuan, an increase of 1.3 times. Growing in tandem with the population pressure is the pressure on urban transportation and travel. Although since October 2008, Beijing has implemented traffic restriction measures, requiring motor vehicles within the fifth ring road to restrict traffic to one day a week according to the tail number, but the road traffic travel situation is still not optimistic. Baidu Maps released the "2023 Annual China City Traffic Report", which shows that in China's 2023 peak commuter traffic congestion index city rankings, Beijing ranked first with a congestion index of 2.125, which increased by 20.13% compared to 2022, and the actual speed of the peak commuter speed of 24.26km/h. In addition to this, Beijing's 2023 commute time consumption and commute distance both ranked the average commuting time consumption is 44.47min and the average commuting distance is 12.53km, and both of them have increased from the previous year. This means that Beijing's road traffic and travel planning, especially road congestion during peak commuting hours, is still a top priority for easing traffic pressure and reducing commuting costs.

In this paper, the road Xizhimen North Avenue in Beijing will be selected as a geographic study area to analyze the congestion prediction of road traffic in the morning peak hour. Xizhimen North Avenue (Figure 1) is located near the Second Ring Road in Beijing, in the area north of the intersection of the West Second Ring Road and the North Second Ring Road, and is one of the important roads in the Xizhimen transportation hub. Near this section of the road is Xizhimen Station, a subway station that ranks among the top ten in terms of passenger flow in Beijing. Located under Xizhimen Bridge in Xicheng District, Beijing, the station is an interchange station for Lines 2, 4, and 13, and it can be directly accessed to the Beijing North Railway Station, which has a very high passenger density, with an average daily passenger flow of up to 130,000 people. In addition, there are a large number of colleges and research institutes in the vicinity of the roadway, such as Beijing Jiaotong University, Beijing University of Posts and Telecommunications, Beijing Normal University, Central University of Finance and Economics, etc., as well as residential areas, and there are a number of large shopping malls and other entertainment venues in the vicinity, which is a crowded place. Therefore, the traffic flow of this roadway is relatively large, especially in the morning and evening peak hours are more congested, which is of research value.

In this study, the section of Xizhimen North Avenue from Xizhimen to Wenhuiqiao is taken as the object of study, and the average speed of vehicles during the morning peak hour (7-10 a.m.) of each day of the second half of 2019 in this road section is taken as the target variable, according to which the degree of congestion is judged, and the data of passenger flow in each hour of the Xizhimen subway station and the meteorological data in the area such as: temperature, air pressure, humidity, precipitation, and other external variables are taken as the characteristic Data, using four time series model algorithms, SARIMA, SVR, GRU, LSTM, to establish a multivariate traffic flow prediction model, and use a variety of model evaluation methods to measure the model performance. This study can further confirm the value of the important influence of external information on traffic flow, provide a scientific basis for the traffic management department to formulate more effective traffic diversion and management strategies, alleviate traffic pressure, and improve traffic efficiency.

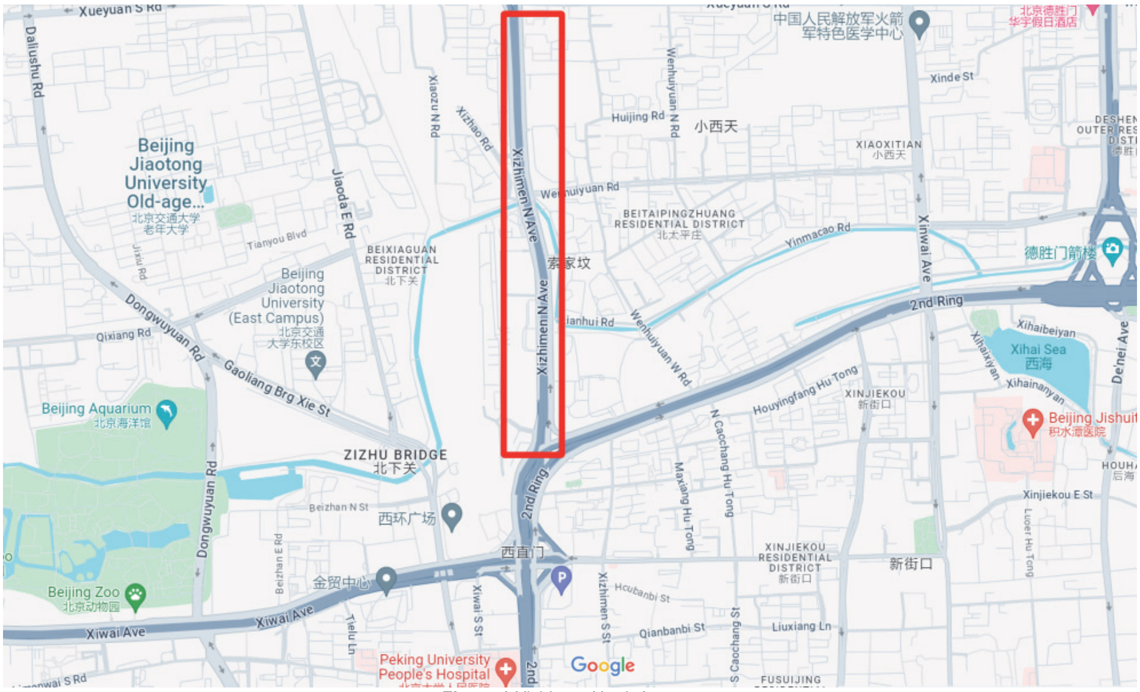


**Figure 1** Xizhimen North Avenue

## 2 RELATED WORKS

### 2.1 Traffic Flow Prediction

Traffic flow prediction is the core problem of intelligent transportation systems, and the accuracy of its prediction directly affects the efficiency of traffic management, urban planning and road navigation [6]. Accurate traffic flow prediction can provide decision support for traffic management, optimize traffic signal control, improve road access efficiency, and reduce traffic congestion and environmental pollution [7]. Therefore, urban road traffic flow prediction has become a research hotspot in academia and industry in recent years.

Since traffic flow data are usually collected in chronological order with obvious time-dependent and cyclical characteristics, the commonly used data in this research field are time series data, and the commonly used analysis methods are also related to time series analysis [8]. Generally speaking, the existing traffic flow prediction models can be divided into three categories: parametric techniques, machine learning techniques, and deep learning techniques [9]. Traditional traffic flow prediction models are mostly parametric methods [10], such as ARIMA, Kalman Filter, and Regression. Although their structure is simple and easy to implement, the prediction accuracy will be reduced due to the assumption limitations and poor adaptability of parametric models [11]. Therefore, many scholars choose to introduce other models to make up for the defects of parametric models in order to optimize the model performance. Xianfu Lin et al. [12] proposed a short-term high-speed traffic flow prediction method based on the arama-garch-m model, which is able to obtain the corresponding fluctuation characteristics relative to the ARIMA model, and improve the prediction accuracy effectively. Siroos Shahriari et al. [13] proposed the E-ARIMA method, which combines the bootstrap method with the ARIMA model so as to form the model ensemble, and the experimental results show that the model ensemble can improve the prediction accuracy.

However, parametric methods can effectively deal with linear relationships and short-term trends, but there are limitations in dealing with complex traffic flow data with nonlinear characteristics [14]. Currently, more scholars choose to use machine learning models for research, and the model prediction effect is significant. SVM is an important machine learning method, but due to the limitations of its binary classification, it is more often chosen to use the SVR model in the prediction of traffic flow to predict continuous variables such as the average speed. Cong Li et al. [15] used CGA to generate the initial population of the data, which was then used as the initial parameters of the SVR model, and used the improved SVR model for short-term prediction of morning and evening peak traffic flow, and obtained a significant reduction in the MAPE and RMSE of the improved model, which has better prediction effect; Haibo Xu et al. [16] combined DBN with SVR to enhance the feature learning ability of the model, so as to improve the accuracy of short-term traffic flow prediction. KNN is also a commonly used machine learning method, which can classify or regress data points according to their proximity in feature space [17]. Lingru Cai et al. [18] proposed a *k*-nearest neighbor regression model with sample rebalancing and outlier rejection to solve the imbalance problem and the problem of limited size of training samples, and the experimental results proved that this model outperforms some of the single-parameter or nonparameter models; Dongwei Xu et al. [19] used the kernel KNN algorithm to establish the road traffic running feature reference sequence (RTRCRS), and mapped the road traffic data sequences in the time series to the feature space to realize the effective prediction of road traffic conditions.

In addition, with the improvement of computing power and the development of big data technology, deep learning methods such as CNN, LSTM, GCN, Encoders, etc. are gradually introduced into the field of traffic flow prediction. Given the strong feature extraction capability and learning training ability of such models, the model prediction accuracy has been further improved. Due to the time series characteristics, the most widely used models are CNN models, LSTM models and their variants [20-22]. In recent years, many scholars are no longer limited to the existing modeling methods, but select a variety of models with different advantages and build a mixture of novel models for traffic flow prediction. Saiqun Lu et al. [23] proposed a combined short-term traffic flow prediction method based on autoregressive integral moving average (ARIMA) model and long-short-term memory (LSTM) neural network, which utilizes a rolling regression ARIMA model to capture the linear regression features of traffic data to optimize the performance of LSTM model; Aniekan Essien et al. [24] used deep bidirectional Long Short-Term Memory (LSTM) stacked self-encoder (SAE) architecture model for multi-step traffic flow prediction on tweets, traffic information and weather datasets, and the results showed that the prediction accuracy of this model was improved; YIQUN LI et al. [25] proposed a hybrid deep learning model W-CNN-LSTM based on wavelet decomposition, convolutional neural network and long and short term memory neural network, which can capture and learn the long term temporal features of the traffic flow well; Mengzhang Li et al. [26] proposed a new spatio-temporal fusion graph neural network for traffic flow prediction (STFGNN), which effectively solves the problem that complex spatio-temporal dependencies are difficult to learn by processing different spatio-temporal maps in parallel.

## 2.2 The SARIMA Model

Given the adaptive and cyclical nature of traffic flow data, S. V. Kumar and L. Vanajakshi proposed seasonal ARIMA model (SARIMA) in 2015 [27]. This model is based on the ARIMA model by adding the structure of seasonal variation, which can effectively deal with time series data with periodic nature.

Let's first review the ARIMA model. The ARIMA model (Autoregressive Integrated Moving Average Model) is a time series forecasting model that consists of three components, the autoregressive model (AR), the differencing process (I), and the moving average model (MA).

The AR component is used to deal with the autoregressive part of the time series, which takes into account the effect of observations from a number of past periods on the current values.

$$Y_t = c + \varphi_1 Y_{t-1} + \varphi_2 Y_{\partial} + \ldots + \varphi_p Y_{t-p} + \xi_t$$

where $Y_t$ denotes the time series data, $\varphi$ describes the relationship between the current value and the value at the past $p$ time points, and $c$ is a constant term. The AR part indicates that the current value is correlated with the past value.

The MA part is used to process the moving average portion of the time series, which takes into account the effect of past forecast errors on the current value.

$$Y_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_{\partial} \varepsilon_{t-2} + \ldots + \theta_q \varepsilon_{t-q}$$

where $Y_t$ denotes the time series data, $\theta$ describes the relationship between the current value and the error at the past $q$ time points, and $\varepsilon_t$ is the error term at time point $t$. The MA part then indicates that the current value is related to the past error term.

Part I is used to smooth the non-stationary time series by reducing or eliminating the trend and seasonality in the time series through multiple difference processing.

The first order difference can be expressed as:

$$\Delta Y_t = Y_t - Y_{t-1}$$

The resulting new series is the difference between every two neighboring values. And the differenced series is somewhat weakened in terms of trend and seasonal variation compared to the original series.

The second order difference can be expressed as:

$$\Delta^2 Y_t = \Delta\left(Y_t - Y_{t-1}\right) - \left(Y_{t-1} - Y_{t-2}\right) = Y_t - 2Y_{t-1} + Y_{t-2}$$

and so on, being able to perform multiple differencing operations based on the ACF and PACF of the data until the sequence is relatively smooth.

And SARIMA adds seasonal component ($S$) compared to ARIMA, including four parameters, seasonal autoregressive order ($P$), seasonal moving average order ($Q$), number of seasonal differences ($D$), and seasonal length ($S$). Among them, the seasonal autoregressive part

($P$) indicates that the past $P$ seasonal data points are used to predict the current value; the seasonal difference part ($D$) indicates that the seasonal component of the time series is smoothed by $D$ times of seasonal difference; the seasonal moving average part ($Q$) indicates that the past $Q$ seasonal error terms are used to predict the current value; and the seasonal period length ($S$) is the seasonal cycle length, which is capable of being expressed in a detailed way by the above parameters detail the seasonal component of the time series. The SARIMA model $SARIMA\left(p,d,q\right)\left(P,D,Q\right)s$ is expressed by the formula as follows:

$$\Phi_P\left(B^s\right)\phi_p\left(B\right)\left(1-B^s\right)^D\left(1-B\right)^d y_t=\Theta_Q\left(B^s\right)\theta_q\left(B\right)\varepsilon_t$$

where $\Phi_P\left(B^s\right)$ is a seasonal autoregressive polynomial, $\phi_P(B)$ is a non-seasonal autoregressive polynomial, $(1-B^s)^D$ and $(1-B)^d$ are seasonal and non-seasonal difference operations, respectively, $\Theta_Q(B^s)$ and $\theta_q(B)$ are seasonal and non-seasonal moving average polynomials, respectively, and $\varepsilon_t$ is a white noise error term.

### 2.3 The SVR Model

Support Vector Regression (SVR, Support Vector Regression) [28] is a regression method based on Support Vector Machine (SVM, Support Vector Machine). Unlike SVM, a machine learning algorithm used for binary categorization, the goal of SVR regression task is to predict continuous numerical outputs rather than discrete category labels. The basic idea of SVR is to solve the regression problem by approximating the given training data by a regression function that fits the data points as closely as possible and at large intervals, and which should have as low a complexity as possible, while ensuring that the prediction error does not exceed a specified threshold (called the ε-insensitive loss). The optimization objective of the SVR model is to minimize the error term ($\min\limits_{w,b}\frac{1}{2}\|W\|^2$) while limiting the error term to no more than $\varepsilon$ ($|\mathrm{y}-\left(wx+b\right)|\le\varepsilon$). Among them, the common kernel trick functions (kernel) mainly include linear kernel, polynomial kernel, Gaussian radial basis kernel (RBF), sigmoid kernel, etc. Different kernel functions can effectively deal with the nonlinear problems of different types of data sets.

The optimization objective of the SVR model is to minimize the error term ($\min\frac{1}{2}\|W\|$) while limiting the error term to no more than $\varepsilon$ ($|\mathrm{y}_i-\left(wx_i+b\right)|\le\varepsilon$).

### 2.4 The LSTM Model

Recurrent Neural Network (RNN) is a type of neural network used to process sequence data, which can combine the information of neighboring nodes to identify that node, and thus be able to learn the long-term and short-term dependencies of each part of the sequence [29]. However, traditional RNNs encounter the problems of gradient vanishing and gradient explosion when dealing with long sequences, making it difficult for the network to learn the long-term dependencies in the sequence. In contrast, Long Short-Term Memory (LSTM, Long Short-Term Memory) [30] (shown in Fig. 2) is a special type of recurrent neural network (RNN), which is able to capture long-term dependencies in sequences more efficiently through the introduction of memory cells (cell states) and gate mechanisms (input gates, forgetting gates, and output gates).

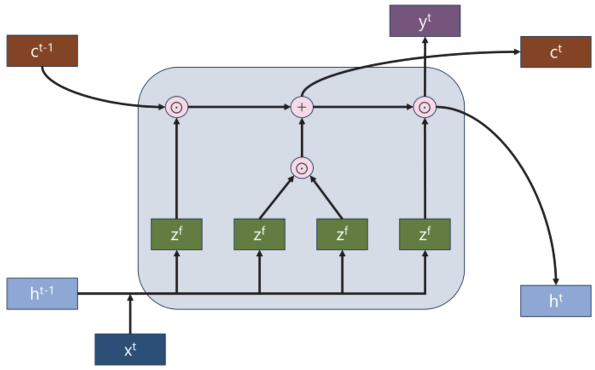


**Figure 2** LSTM structure

In this model, there are two transmitted states, $c^t$ (cell state) and one $h^t$ (hidden state) where $x^t$ is the input of the current neuron, $h^{t-1}$ and $c^{t-1}$ are the inputs transmitted from the previous state.

There are three main phases within the LSTM model:

**Step1:** Forgetting stage. This stage requires selective forgetting of the information entered in the previous stage, using the forgetting gate $z^f$ to determine which parts of the current unit state need to be forgotten, which is calculated by the formula: $z^f=\sigma\left(W^f\left[x^t,h^{t-1}\right]\right)$, and applying the calculated $z^f$ to control which information needs to be forgotten in the $c^{t-1}$ passed down from the previous state.

**Step2:** Selective Memorization Stage. This stage requires selective memorization of the input $x^t$ in this stage, focusing on memorizing important information. The current input is the value $z$ converted using the tanh activation function with the formula $z=\tanh\left(W\left[x,h\right]\right)$. and controlled by the selection gating signal $z^i$ with the formula $z^i=\sigma\left(W^i\left[x^t,h^{t-1}\right]\right)$.

**Step3:** Output stage. This stage requires the output gating $z^o$ to control the output information of this unit, which is calculated by the formula $z^o=\sigma\left(W^o\left[x^t,h^{t-1}\right]\right)$. The calculation formula of the final output information is expressed as follows:

$$c^t=z^f\odot c^{t-1}+z^i\odot z$$

$$h^t=z^o\odot\tanh\left(c^t\right)$$

$$y^t=\sigma\left(W'h^t\right)$$

### 2.5 The GRU Model

Similar to LSTM, GRU model [31] is also a special kind of RNN model, but unlike it, the GRU model is relatively more simplified, with only two gating controls, the reset gate and the update gate (Fig. 3). As a result, the GRU model has fewer variables than LSTM, the model parameters are easier to adjust, and the training efficiency will be higher.

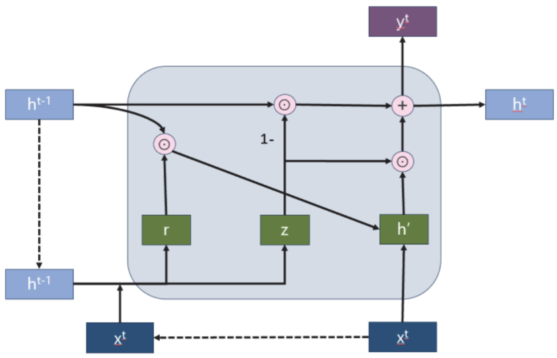


**Figure 3** GRU structure

where $r$ and $z$ are reset gating and update gating, respectively, computed as $r=\sigma\left(W^r\left[x^t,h^{t-1}\right]\right)$, $z=\sigma\left(W^z\left[x^t,h^{t-1}\right]\right)$. The GRU model is internally divided into two main phases: reset and update memory:

Step1: Reset stage. First use the reset gating to get the reset data $h^{t-1}{}'=h^{t-1}\odot r$, then combine it with the model input of this layer $x^t$ to get $h'=\tanh\left(W\left[x^t,h^{t-1}{}'\right]\right)$.

Step 2: Update Memory Phase. Different from the LSTM model, the GRU model will forget and memorize at the same time in this stage, by $(1-z)\odot h^{t-1}$ denoting the forgetting of the information passed down from the previous layer, and by $z\odot h'$ denoting the memorizing of the important information in the current node, the final input expression is obtained as:

$$h^t=(1-z)\odot h^{t-1}+z\odot h'$$

$$y^t=\sigma\left(W'h^t\right)$$

## 3 EXPERIMENTS AND RESULT ANALYSIS

### 3.1 Data Source

The data of this study is divided into four datasets, which are the traffic flow dataset of Xizhimen North Avenue (Wenhui Bridge to Xizhimen) road section in Beijing every day from 7-10 a.m. on an hourly basis from July 1 to December 31, 2019 for this road section, the holiday dataset, the passenger flow dataset of the Xizhimen subway entrance, and the meteorological dataset, and the content of the specific fields is as shown in the Tab. 1.

**Table 1** Description of datasets fields

| Dataset | Feature | Description |
|---|---|---|
| Traffic | Datetime | timing |
| | Year | particular year |
| | Month | months |
| | Date | dates |
| | Hour | hourly |
| | Fluid | free-flow |
| | Speed | Average vehicle speed / km/h |
| Holiday | Week | weeks |
| | Holiday | Whether it is a holiday |
| | Weekday | Whether it is a working day |
| Subway | Subway | Passenger flow at metro stations (person trips) |
| Weather | Humid | Air humidity / % |
| | Pressure | Air pressure / hPa |
| | Precipitation | Precipitation / mm/h |
| | Windspeed | Wind speed / m/s |
| | Temperature | Temperature / °C |

### 3.1 Data Processing

First of all, basic clear operations are carried out on the data, such as: deleting duplicate values, filling in missing items, etc.; and the time data are refined by adding the information of month, day, hour, whether it is a working day or not, and whether it is a legal holiday, etc., into the dataset as new fields, so as to enhance the degree of influence of the time characteristics on the training of the model. A heat map of the correlation between the factors (Fig. 4) was established to initially understand the correlation between the factors and determine the target variables.

According to the preliminary correlation assessment, combined with the purpose of the study, the average speed (speed) is selected as the target variable, and a random forest model is established to calculate the importance of the influence of each field on speed, in order to screen out the feature indicators with a higher degree of influence on the target variable for subsequent modeling. The calculated importance of each characteristic factor is shown in the table below, and finally week, hour, festival, weekday, subway, humid, pressure, precipitation, windspeed, temperature are selected as the subsequent research variables.

During the study, the first 80% (147 days) of a total of 736 data (184 days) of this dataset was used as a training set for model training, and the second 20% (37 days) was used as a test set for testing the model performance, with a total of 588 pieces of data used for model training and 148 pieces of data used for model testing and evaluation.

**Table 2** Characteristic importance values

| NO. | Feature | Importance | NO. | Feature | Importance |
|---|---|---|---|---|---|
| 1 | Week | 0.270461 | 8 | Date | 0.033806 |
| 2 | Hour | 0.199582 | 9 | Temperature | 0.029838 |
| 3 | Weekday | 0.183444 | 10 | Windspeed | 0.028102 |
| 4 | Holiday | 0.114165 | 11 | Month | 0.008206 |
| 5 | Subway | 0.050841 | 12 | Precipitation | 0.007713 |
| 6 | Pressure | 0.034081 | 13 | Fluid | 0.005706 |
| 7 | Humid | 0.034053 | 14 | year | 0.000000 |

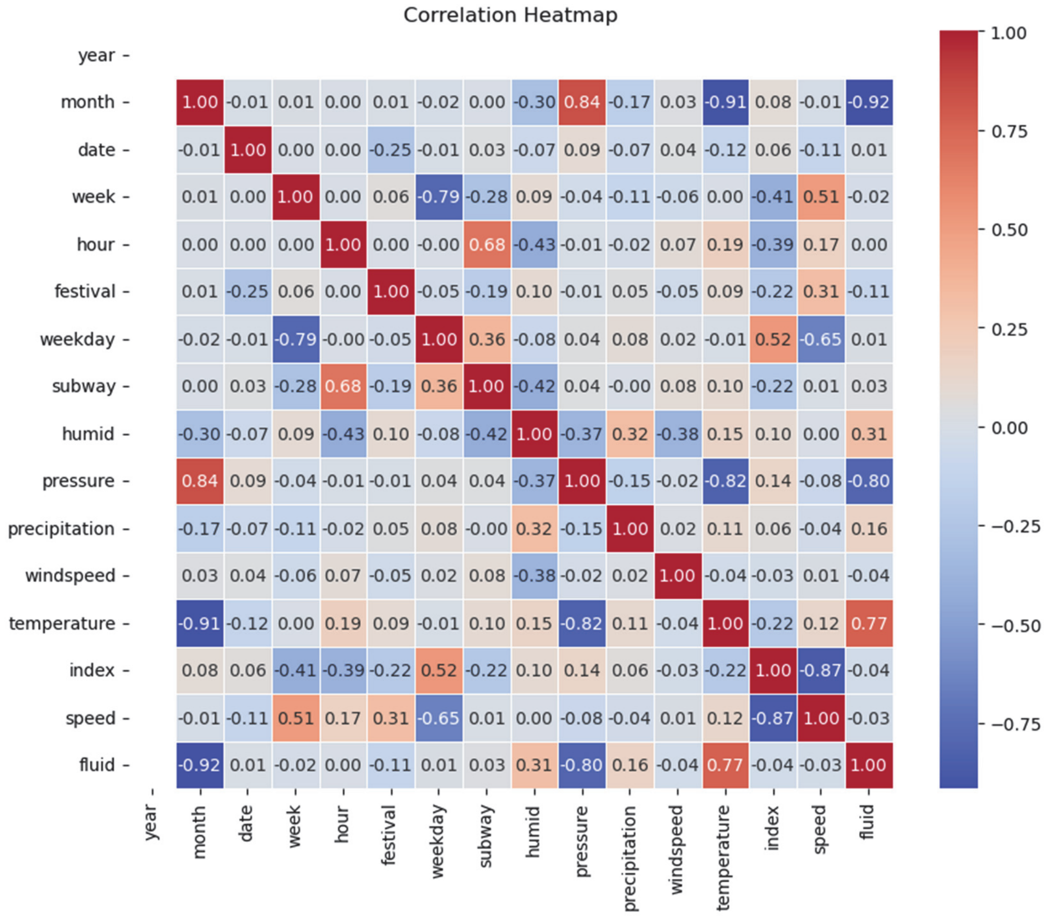


**Figure 4** Heat map of correlation of variables

## 3.2 Model Training

Currently, urban traffic flow prediction techniques can be broadly divided into three categories: parametric techniques, machine learning and deep learning [9]. Among them, the representative modeling methods in the Parametric techniques type are ARIMA, Kalman Filter, Regression, etc., the representative models in the machine learning category are ANN, KNN, SVM, and the most frequently used deep learning methods are CNN, LSTM, GRU, GCN, Encoders, etc. In this study, one or two models from each of these three classes of modeling methods are chosen to broaden the scope of the study.

### 3.2.1 SARIMA Model

SARIMA is a time series model, an upgraded version of ARIMA (Autoregressive Integral Sliding Average Model) with a seasonal component, which is suitable for time series data with significant seasonality and trend, and is able to analyze the seasonality by providing accurate long-term forecasting and seasonality.

From decomposition of the trend and periodicity of the data (as shown in Fig. 5), it can be seen that the data has a clear periodicity, so we consider the use of the SARIMA model, which has a better prediction effect on the periodic data, for modeling. According to the time characteristics of the data and the periodicity presented, it is determined that the period of the data is 7 (days) × 4 (hours) = 28, i.e., 28 pieces of data for a cycle.

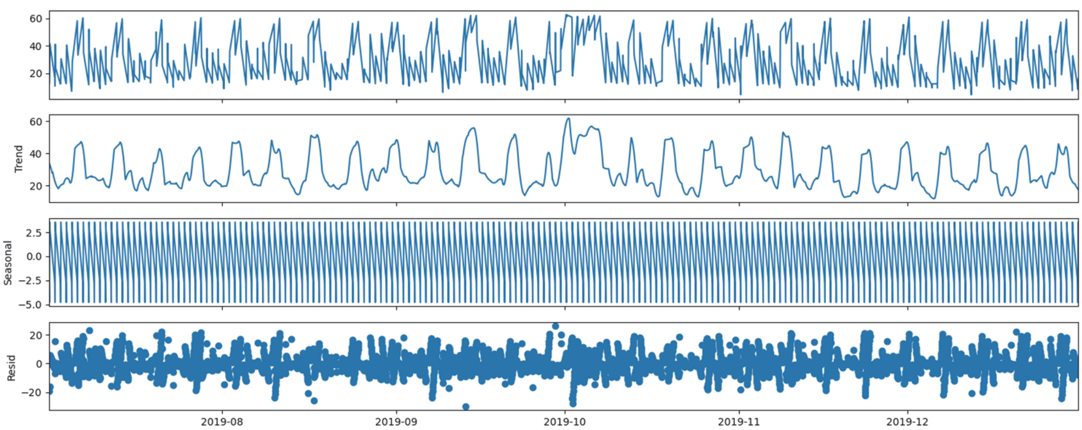


**Figure 5** Decomposition of seasonal cyclical trends

Seasonal differencing was first performed on the data to remove the seasonality of the data in terms of period 28. Based on the autocorrelation and partial autocorrelation order of the data at this point, it was decided to perform another differencing operation, and the final data and its autocorrelation and partial autocorrelation were obtained as shown in figure 6. The final parameters $p = 3$, $d = 1$, $q = 2$, $P = 0$, $D = 1$, $Q = 1$, $S = 28$ were determined to build the SARIMA model and MSE, MAE, RMSE were calculated to assess the model performance.

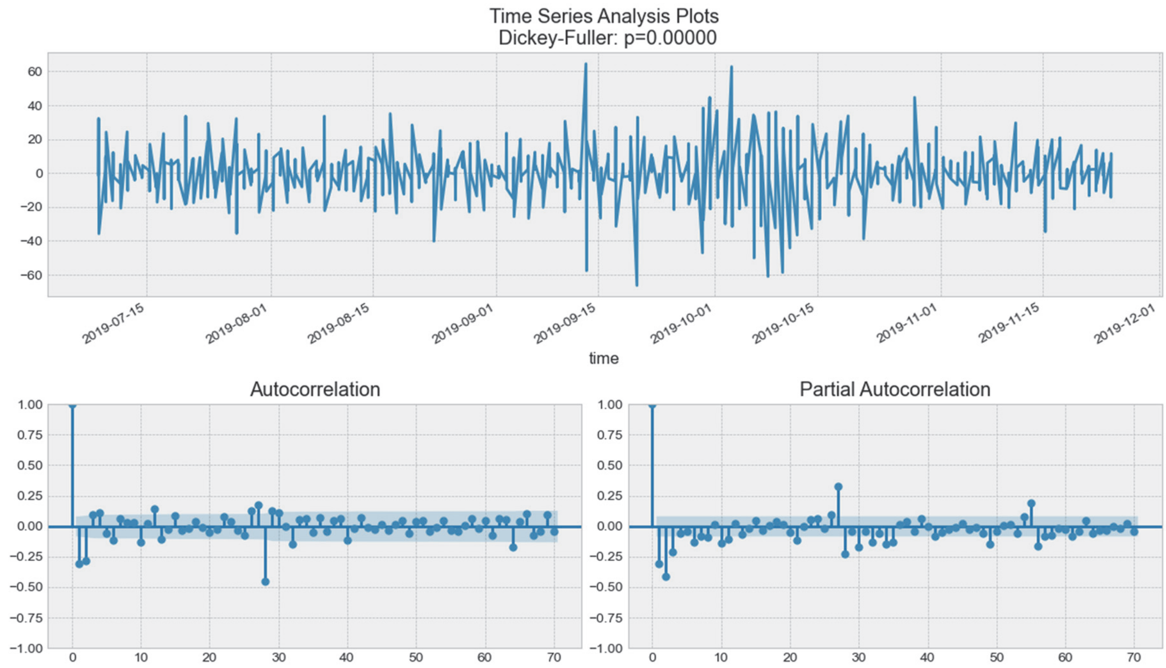


**Figure 6** Time series diagnostic chart

### 3.2.2 SVR Model

SVR is a regression method based on support vector machine, which is able to use different kernel functions to find the best regression planes or curves in the data, and perform nonlinear regression analysis in high-dimensional space, and especially perform well in dealing with nonlinear and complex data patterns.

After loading the data, the variables identified by previous feature engineering were first selected as experimental data, and after dividing the training and test sets and their independent and dependent variable data, the field datetime (YYYY-MM-DD HH:00:00) was set as the index column of the time series data. After that, MinMaxScaler() is applied to normalize the numerical data to unify the scale difference of different types of data, accelerate the convergence to improve the training speed, and reduce the numerical instability to improve the robustness of the model.

After that, the candidate types of kernel function are set as 'rbf', 'poly', 'sigmoid', and the candidate values of regularization parameter are 1,10,100, and grid search is applied to calculate the optimal model parameters. The optimal regularization parameter (C) is 1, 'kernel': the optimal kernel is radial basis function kernel (RBF), and the optimal coefficient gamma of RBF is 0.1. Based on this optimal parameter, the SVR model is built on the training set, and the multi-featured variables in the test set are used to predict the target variable speed, and the MSE, MAE, MAE, and speed, are calculated based on the predicted values and the true values. MSE, MAE, RMSE are calculated to evaluate the model performance.

### 3.2.3 GRU Model

GRU is a simplified version of recurrent neural network (RNN) with a simpler structure containing update and reset gates, which is more computationally efficient, and is suitable for sequential data tasks requiring memorization of long term dependencies, and is able to provide faster training and higher prediction accuracy while retaining long term dependency information.

Firstly, the numerical data are normalized and the training set and test set data are created, and the feature dataset is constructed with a sliding window of data period 28, which generates a total of training feature dataset with a capacity of 560 items and a test feature dataset with a capacity of 120 items. After constructing the batch data with a batch size of 126, a GRU model is built for the training feature data set.

After several parameter adjustments, the finalized GRU model structure is shown in the following figure: the model has a three-layer structure, the first layer of the GRU contains 256 hidden layer units, the second and third layers contain 128 and 64 hidden layer units, respectively, and the Dropout layer rate between the first layer and the second layer is 0.5, and the Dropout rate between the second layer and the third layer is 0.3, respectively, indicating that Each time the parameters are updated a corresponding percentage of input neuron connections is randomly disconnected to prevent overfitting, and finally a single value is output for regression prediction. The model uses the Adam optimizer with a learning rate of 0.001 and the loss function is calculated using the Mean Square Error (MSE) to assess whether the model is fit or not, and the number of iterations is set to 250. Afterwards, the model is applied to predict the data in the test set and the error metrics between the predicted and the true value of the target variable and the degree of model fit are calculated to measure the performance of the model.

#### 3.2.4 LSTM Model

The LSTM Model is a special kind of recurrent neural network (RNN) designed to capture and retain long time dependency information, overcoming the gradient vanishing problem in traditional RNNs, suitable for sequence data tasks that need to memorize long time dependencies, able to effectively retain and transfer long time dependency information through gating mechanism, LSTM can provide high accuracy in complex sequence prediction tasks.

Firstly, the numerical data are normalized and the training set and test set data are established, and the feature dataset is constructed with a sliding window of data period 28, and a total of training feature dataset with a capacity of 560 items and a test feature dataset with a capacity of 120 items are generated. After constructing the batch data with a batch size of 126, an LSTM model is built for the training feature data set.

After several parameter adjustments, the finalized LSTM model structure is shown in the following figure: the model has three LSTM layers, the first LSTM contains 256 hidden layer units, the second LSTM contains 128 hidden layer units, the third LSTM contains 64 hidden layer units, and there is a Dropout layer between every two LSTM layers, and the Dropout rates are respectively 0.4 and 0.2, indicating that the corresponding percentage of input neurons is randomly disconnected each time the parameters are updated to prevent overfitting, and the final output of a single value is used for regression prediction. The model uses the Adam optimizer with a learning rate of 0.001 and the loss function is calculated using the Mean Square Error (*MSE*) to assess whether the model is fit or not, and the number of iterations is set to 240. After that, the model is applied to predict the data in the test set and the error metrics between the predicted and the true value of the target variable and the degree of model fit are calculated to measure the performance of the model.

### 3.3 Results Analysis

The experimental results of each model are analyzed, including the interpretation and comparative analysis of the evaluation indicators and the performance of the model in actual prediction. The experimental results analysis section evaluates and comparatively analyzes the prediction results of the four models.

#### 3.3.1 Prediction Results

The predicted and true values of the models for the test set are compared and the corresponding line graphs are plotted as shown in Fig. 7. From the model prediction results, all four models are able to predict the dataset to a certain extent, with a certain fitting ability. However, in contrast, the SVR model has the weakest fitting effect compared with the other three models, and although it can basically determine the trend of data changes, the prediction accuracy is too low, and the deviation between the predicted value and the real value is large. In contrast, the two deep learning models - GRU and LSTM - have the best fitting, and are able to predict the data trends and values more accurately.

#### 3.3.2 Selection of the Evaluation Function

$R^2$ (*R*-Squared): $R^2$ is the coefficient of determination, which indicates how well the model explains the variance of the data. Its value ranges from 0 to 1. The closer it is to 1, the better the model explains the data variance, i.e., the better the model fits.

$$R^2 = 1 - \frac{\sum_t \left(\hat{y}_t - y_t\right)^2}{\sum_t \left(\bar{y}_t - y_t\right)^2}$$

*MSE* (Mean Squared Error): *MSE* is the average of the squares of the differences between the predicted values and the true values, which is a common loss function. The smaller the MSE is, the smaller the gap between the predicted and the true values of the model.

$$MSE = \frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2$$

*RMSE* (Root Mean Squared Error): *RMSE* is the square root of *MSE*, which has the same unit of measure as *MSE* but is easier to interpret. The smaller the *RMSE* is, the smaller the prediction error of the model.

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}\left(y_t - \hat{y}_t\right)^2}$$

*MAE* (Mean Absolute Error): *MAE* is the average of the absolute value of the difference between the predicted value and the true value, which indicates the average deviation between the predicted value and the true value. The smaller the *MAE* is, the smaller the average prediction error of the model is.

$$MAE = \frac{1}{n}\sum_{t=1}^{n}|\, y_t - \hat{y}_t \,|$$

The results of multiple assessment metrics calculated between the predicted and true values of the four models are obtained as shown in the table. Since the SARIMA model is a statistical model used for time series forecasting, and *R*2 is a commonly used evaluation metric for linear regression models to measure the extent to which the model explains the variance of the observed data, the SARIMA model does not usually use *R*-squared ($R^2$) directly as an evaluation metric.

As can be seen from the data in the Tab. 3, the LSTM model has the best overall performance, both in terms of the degree of model fit and error, with an $R^2$ value of more than 0.8, and its generated predictions are able to match the real values of the test set to a greater extent and with a smaller error. Similar to the LSTM results, the GRU model has a similar gating mechanism, which deals with long-term dependence by selectively remembering and forgetting information. The GRU model has the second best performance after the LSTM model, with an $R^2$ value close to 0.8, and its predicted values have a smaller error with the true values, both less than 1. Both models can effectively capture long-term dependence and complex nonlinear relationships, and are particularly good at dealing with time series data. They are especially good at dealing with time series data, while their gating mechanism allows them to retain important temporal information while gradually discarding irrelevant information, and thus are able to achieve better performance on this dataset.

**Table 3** Indicators for model evaluation

| Model | R2 | MAE | MSE | RMSE |
|---|---|---|---|---|
| SARIMA | | 5.65 | 52.04 | 7.21 |
| LSTM | 0.823 | 0.0899 | 0.014 | 0.1185 |
| SVR | 0.456 | 0.17 | 0.042 | 0.21 |
| GRU | 0.794 | 0.098 | 0.016 | 0.128 |

This is because the SARIMA model is more suitable for dealing with linear time series data with seasonality and trend, but it does not perform as well as the deep learning model in dealing with complex nonlinear and long time-dependent problems, and its simple structure limits its performance on complex time series. The $R^2$ value of SVR model is 0.456, which indicates that it can only explain a part of the variance in the data and performs relatively poorly. Moreover, the MAE and RMSE are also higher, indicating that SVR has a higher prediction error on this data set. This also proves that the SVR model is not as effective as deep learning models such as LSTM and GRU in dealing with nonlinear time series data, especially on datasets with long time dependency.

## 4 CONCLUSION

In this paper, four methods, namely SARIMA, SVR, GRU, and LSTM, are chosen to predict urban road traffic flow. It is worth mentioning that the feature variables chosen in this study are all other types of variables besides traffic factors, such as passenger flow in nearby subway stations, holiday information, and meteorological data, which further corroborate the degree of influence of external factors on traffic flow.

The results of the study show that the model containing only external variables can effectively predict the road congestion condition. Among them, the best prediction results are the LSTM model and the GRU model, both of which have an R2 value as high as 0.79, and the MAE and MSE values are less than 0.1, indicating that the predicted values obtained by the model have a high degree of match with the real values. In contrast, the SVR model has a more average prediction performance, with an $R^2$ value of 0.456 and MAE and MSE values of 0.17 and 0.042, respectively. It is not as accurate as the LSTM and GRU models, despite its ability to capture trends. The SARIMA model, on the other hand, performs the worst, and its evaluation metrics (e.g., MAE, MSE, and RMSE) show that it has a large prediction error, which may be attributed to the fact that the SARIMA model is more suitable for linear and seasonally significant data, while it has limited effectiveness in predicting the complex nonlinear traffic flow.

In summary, this study verifies that external factors have a significant impact on traffic flow by comparing the prediction effects of different models, and also confirms that the deep learning models LSTM and GRU have a significant advantage over the ordinary machine learning model SVR and parametric model SARIMA in dealing with complex and nonlinear traffic flow data. Future research can further optimize these models or combine other external variables and features to improve prediction accuracy and practicality.

## 5 REFERENCES


[1] Cui, J. Y. & Zhao, B. (2024). A review of short-term traffic flow prediction methods based on deep learning. *Journal of Transportation Engineering*, (02), 50-64. https://doi.org/10.19818/j.cnki.1671-1637.2024.02.003

[2] Zhang, L. Y., Duan, X. K., Ma, J., Zhang, M., Wen, Y., & Wang, Y. (2022). Mechanism of road capacity under different penetration scenarios of autonomous vehicles. *International Journal of Simulation Modelling*, *21*(1), 172-183. https://doi.org/10.2507/IJSIMM21-1-CO4

[3] Sayed, S. A., Abdel-Hamid, Y., & Hefny, H. A. (2023). Artificial intelligence-based traffic flow prediction: a comprehensive review. *Journal of Electrical Systems and Information Technology*, *10*(1), 13. https://doi.org/10.1186/s43067-023-00081-6

[4] Hou, Y., Deng, Z., & Cui, H. (2021). Short-Term Traffic Flow Prediction with Weather Conditions: Based on Deep Learning Algorithms and Data Fusion. *Complexity*, *2021*(1), 6662959. https://doi.org/10.1155/2021/6662959

[5] Kashyap, A. A., Raviraj, S., Devarakonda, A., Nayak, K. S. R., KV, S., & Bhat, S. J. (2022). Traffic flow prediction models-A review of deep learning techniques. *Cogent Engineering*, *9*(1), 2010510. https://doi.org/10.1080/23311916.2021.2010510

[6] Akhtar, M. & Moridpour, S. (2021). A review of traffic congestion prediction using artificial intelligence. *Journal of Advanced Transportation*, *2021*, 1-18. https://doi.org/10.1155/2021/8878011

[7] Katambire, V. N., Musabe, R., Uwitonze, A., & Mukanyiligira, D. (2023). Forecasting the Traffic Flow by Using ARIMA and LSTM Models: case of Muhima Junction. *Forecasting*, *5*(4), 616-628. https://doi.org/10.3390/forecast5040034

[8] Méndez, M., Merayo, M. G., & Núñez, M. (2023). Long-term traffic flow forecasting using a hybrid CNN-BiLSTM model. *Engineering Applications of Artificial Intelligence*, *121*, 106041. https://doi.org/10.1016/j.engappai.2023.106041

[9] Medina-Salgado, B., Sánchez-DelaCruz, E., Pozos-Parra, P., & Sierra, J. E. (2022). Urban traffic flow prediction techniques: a review. *Sustainable Computing: Informatics and Systems*, *35*, 100739. https://doi.org/10.1016/j.suscom.2022.100739

[10] Zheng, H., Lin, F., Feng, X., & Chen, Y. (2020). A hybrid deep learning model with attention-based conv-LSTM networks for short-term traffic flow prediction. *IEEE Transactions on Intelligent Transportation Systems*, *22*(11), 6910-6920. https:// doi.org/10.1109/TITS.2020.2997352

[11] Zhao, L., Wen, X., Wang, Y., & Shao, Y. (2022). A novel hybrid model of ARIMA-MCC and CKDE-GARCH for urban short-term traffic flow prediction. *IET Intelligent Transport Systems*, *16*(2), 206-217. https://doi.org/10.1049/itr2.12138

[12] Lin, X. & Huang, Y. (2021). Short-term high-speed traffic flow prediction based on ARIMA-GARCH-M model.

*Wireless Personal Communications*, *117*(4), 3421-3430. https://doi.org/10.1007/s11277-021-08085-z

[13] Shahriari, S., Ghasri, M., Sisson, S. A., & Rashidi, T. (2020). Ensemble of ARIMA: combining parametric and bootstrapping technique for traffic flow prediction. *Transportmetrica A: Transport Science*, *16*(3), 1552- 1573. https://doi.org/10.1080/23249935.2020.1764662

[14] Chen, K., Chen, F., Lai, B., Jin, Z., Liu, Y., Li, K., Wei, L., Wang, P., Tang, Y., Huang, J., & Hua, X. (2020). Dynamic spatio-temporal graph-based CNNs for traffic flow prediction. *IEEE Access*, *8*, 185136-185145. https:// doi.org/10.1109/ access.2020.3027375

[15] Li, C. & Xu, P. (2021). Application on traffic flow prediction of machine learning in intelligent transportation. *Neural Computing and Applications*, *33*(2), 613-624. https://doi.org/10.1007/s00521-020-05002-6

[16] Lin, G., Lin, A., & Gu, D. (2022). Using support vector regression and K-nearest neighbors for short-term traffic flow prediction based on maximal information coefficient. *Information Sciences*, *608*, 517-531. https://doi.org/10.1016/j.ins.2022.06.090

[17] Cover, T. & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE transactions on information theory*, *13*(1), 21-27. https:// doi.org/10.1109/TIT.1967.1053964T

[18] Cai, L., Yu, Y., Zhang, S., Song, Y., Xiong, Z., & Zhou, T. (2020). A sample-rebalanced outlier-rejected $ k $-nearest neighbor regression model for short-term traffic flow forecasting. *IEEE access*, *8*, 22686-22696. https://doi.org/10.1109/access.2020.2970250

[19] Xu, D., Wang, Y., Peng, P., Beilun, S., Deng, Z., & Guo, H. (2020). Real-time road traffic state prediction based on kernel-KNN. *Transportmetrica A: Transport Science*, *16*(1), 104-118. https://doi.org/10.1080/23249935.2018.1491073

[20] Doğan, E. (2021). LSTM training set analysis and clustering model development for short-term traffic flow prediction. *Neural Computing and Applications*, *33*(17), 11175-11188. https://doi.org/10.1007/s00521-020-05564-5

[21] Wang, S., Zhao, J., Shao, C., Dong, C., & Yin, C. (2020). Truck traffic flow prediction based on LSTM and GRU methods with sampled GPS data. *IEEE Access*, *8*, 208158-208169. https://doi.org/10.1109/access.2020.3038788

[22] Wang, Z., Su, X., & Ding, Z. (2020). Long-term traffic prediction based on lstm encoder-decoder architecture. *IEEE Transactions on Intelligent Transportation Systems*, *22*(10), 6561- 6571. https://doi.org/10.1109/tits.2020.2995546

[23] Lu, S., Zhang, Q., Chen, G., & Seng, D. (2021). A combined method for short-term traffic flow prediction based on recurrent neural network. *Alexandria Engineering Journal*, *60*(1), 87-94. https://doi.org/10.1016/j.aej.2020.06.008

[24] Essien, A., Petrounias, I., Sampaio, P., & Sampaio, S. (2021). A deep-learning model for urban traffic flow prediction with traffic events mined from twitter. *World Wide Web*, *24*(4), 1345-1368. https://doi.org/10.1007/s11280-020-00800-3

[25] Li, Y., Chai, S., Ma, Z., & Wang, G. (2021). A hybrid deep learning framework for long-term traffic flow prediction. *IEEE Access*, *9*, 11264-11271. https://doi.org/10.1109/access.2021.3050836

[26] Li, M. & Zhu, Z. (2021). Spatial-temporal fusion graph neural networks for traffic flow forecasting. *Proceedings of the AAAI conference on artificial intelligence*, *35*(5), 4189-4196. https://doi.org/10.1609/aaai.v35i5.16542

[27] Kumar, S. V. &Vanajakshi, L. (2015). Short-term traffic flow prediction using seasonal ARIMA model with limited input data. *European Transport Research Review*, *7*, 1-9. https://doi.org/10.1007/s12544-015-0170-8

[28] Vapnik, V., Golowich, S., & Smola, A. (1996). Support vector method for function approximation, regression estimation and signal processing. *Advances in Neural Information Processing Systems*, *9*.

[29] Tedjopurnomo, D. A., Bao, Z., Zheng, B., Choudhury, F. M., & Qin, A. K. (2020). A survey on modern deep neural network for traffic prediction: Trends, methods and challenges. *IEEE Transactions on Knowledge and Data Engineering*, *34*(4), 1544-1561. https://doi.org/10.1109/tkde.2020.3001195

[30] Hochreiter, S. & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

[31] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078. https://doi.org/10.3115/v1/d14-1179

**Contact information:**

**Jichen WANG**
School of Economics and Management,
Beijing Jiaotong University,
Beijing 100044, China
E-mail: a15137599126@outlook.com