International Journal for Housing Science and Its Applications
Publish August 10, 2025. Volume 46, Issue 4 Pages 4435-4446

https://doi.org/10.70517/ijhsa464289

Attention-Weighted Traffic Flow Prediction and Congestion Early Warning Study with Synergy of ETC Gantry and Internet of Things Monitoring Data

Zhengqiong Wang[1,*]
[1] Yunnan Yuntong Shulian Technology Co., Ltd., Kunming, Yunnan, 650100, China
Corresponding authors: (e-mail: 13518713626@163.com).

Abstract With the increasing demand of urban traffic management, intelligent transportation system (ITS) has gradually become an important means to solve the problem of urban traffic congestion. The combination of ETC gantries and Internet of Things (IoT) monitoring technology provides data support for accurate prediction of real-time traffic flow and congestion warning. In this paper, an attention-weighted SG-LSTM-based traffic flow prediction model is proposed and applied to the traffic flow prediction and congestion warning of ETC gantry data in M city A area. Through data preprocessing, the introduction of Savitzky-Golay filter, and the training of LSTM neural network, this model can effectively improve the accuracy of traffic flow prediction. The experimental results show that the model has higher prediction accuracy compared to the traditional LSTM, CNN, GCN and other classical methods. Specifically, the model reduces 27.43% and 43.07% in RMSE and MAE metrics, respectively. While the accuracy of traffic flow prediction is improved, this study also designs a congestion warning model based on support vector machine, which predicts the traffic flow and speed through real-time data and accurately warns the traffic congestion condition, which verifies the effectiveness and high accuracy of this model.

Index Terms ETC gantry, traffic flow prediction, attention weighting, SG-LSTM, congestion warning, support vector machine

I. Introduction

Traffic management has always been one of the important issues in urban development [1]. With the increase of population and the popularity of transportation, the problems of traffic congestion and frequent traffic accidents are becoming more and more prominent [2]. With the application of Internet of Things (IoT) and ETC gantries, the field of transportation has been intelligentized to enable real-time, accurate and efficient traffic management, which effectively improves transportation efficiency, reduces traffic congestion, lowers energy consumption, improves traffic safety and enhances the quality of travel services [3]-[6].

ETC gantry is a kind of intelligent traffic management means based on ETC system and sensor technology [7]. This technology realizes real-time monitoring and information collection of vehicles by installing ETC and sensors above the highway [8], [9]. This information includes vehicle model, driving speed, driving direction, etc., providing all-round data support for highway operation and management [10], [11]. And through the Internet of Things monitoring data synergies can make the ETC system become smarter, before the vehicle through the toll booths to stop and pay cash, now ETC through the wireless identification of automatic toll deduction, time-saving and convenient [12]-[14]. Of course, the Internet of Things is not simply to install a few sensors to the ETC, but to build up a complete closed loop of sensing, transmission, analysis, response, and each technological improvement should take into account the variables in the actual scenarios, such as extreme weather, equipment aging, human damage and other factors [15]-[18]. The synergistic application of the two can not only charge fees, but also provide value-added services such as traffic flow prediction, congestion warning, and vehicle health monitoring.

In this study, based on the ETC gantry traffic data in M City A, data preprocessing is combined with Savitzky-Golay filter, traffic flow is predicted by SG-LSTM model, and a traffic congestion warning system is designed based on Support Vector Machine (SVM). In the study, the data were first cleaned and smoothed, and then the deep learning model was used for traffic flow prediction and combined with real-time data for congestion prediction. Finally, the validity and high accuracy of the model are verified through experiments, and an intelligent traffic management solution that can predict traffic flow and congestion risk simultaneously is proposed.

II. Experimental parameterization

II. A. Description of the data set

In this paper, the publicly available urban ETC gantry IoT detection dataset is selected, and this dataset is the vehicle trajectory data, traffic flow characteristics data, road network data, and weather data of each ETC gantry in the one-year time interval from January 1, 2024 to December 31, 2024 in City A of M.

The specific data includes 245 road sections, and only the food service, transportation facilities, accommodation service, and medical service in the data set are taken in this experiment due to the overly cumbersome data and insufficient server performance. The service attribute values are represented using the parameters 0 and 1, with 1 indicating that it belongs to this neighborhood. For the weather data, it belongs to the dynamic external factor data, in which the weather conditions are categorized into five categories in this paper: sunny, cloudy, foggy, rainy, and stormy, and the dataset counts the weather information every 20 minutes.

The data of food service, transportation facilities, accommodation service, and medical service are constructed as attribute triples, for example, (road section 1, restaurants, 3) and (moment t, weather, 0) indicate that there are 3 restaurants around road section 1 and that the weather is sunny in Luohu District at moment t, respectively. The input data includes road ternary (head entity, relationship, tail entity), attribute ternary (entity, attribute, attribute value) and attribute co-occurrence probability ternary (attribute 1, attribute 2, attribute co-occurrence probability).

II. B. Experimental environment

The hardware environment for this experiment is: CPU: Intel i7 14700KF, GPU: TX4070Ti, 32GB of RAM, and the software environment is Ubuntu20.04+Python3.8+TensorFlow2.6.0.

II. C. Evaluation indicators

The performance metrics selected for this study are Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Coefficient of Determination (R^2), and Prediction Accuracy (Acc). The mathematical expression formulas for the four performance metrics are shown in the following equations:

$$RMSE = \sqrt{\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(y_i^j - \hat{y}_i^j\right)^2} \tag{1}$$

$$MAE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left|y_i^j - \hat{y}_i^j\right| \tag{2}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\left(y_i^j - \hat{y}_i^j\right)^2}{\sum_{i=1}^{M}\sum_{j=1}^{N}\left(y_i^j - \bar{Y}\right)^2} \tag{3}$$

$$Acc = 1 - \frac{\left\|Y - \hat{Y}\right\|_F}{\left\|Y\right\|_F} \tag{4}$$

where y_i^j is the true speed of the traffic flow of the i road at the moment of j, $\hat{y}_i^j$ is the predicted speed of the traffic flow of the i road at the moment of j, M is the total number of roads, N is the length of the data set and the number of samples collected, Y is the set of y_i^j, $\hat{Y}$ is the set of $\hat{y}_i^j$, and $\bar{Y}$ is the average value of Y.

III. SG-LSTM traffic flow prediction model based on attention weighting

III. A. S-G Filter

Savitzky-Golay (S-G) filter [19] is a filtering method based on local polynomial least squares fitting in the time domain. It can be applied to datasets for data smoothing while keeping the trend and width of the signal constant, thus improving the accuracy of the data. The use of S-G filter reduces the data noise interference, improves the data periodicity and predicts the traffic flow more accurately. The Savitzky-Golay smoothing formula is shown in:

$$\begin{aligned} &\overline{x}_{\xi} = \sum_{r=\frac{1-m}{2}}^{\frac{m-1}{2}} c_{\xi} x_{r+\xi} \\ &s.t.: \frac{m-1}{2} \le \xi \le t - \frac{m-1}{2} \end{aligned} \tag{5}$$

Here, $\overline{x}_{\xi}$ is the smoothed result, m is the filter window size, $x_{r+\xi}$ is the data points in the window, and c_{ξ} is the smoothing coefficient.

A least squares method with m smoothing coefficients is used to estimate the window $x_{r+\xi}(\xi = 1,\dots,T)$, and the window size affects the effectiveness of the S-G filter.

Calculation of the smoothing coefficient C_{ξ}: fit neighboring data points $X = \left\{ x_{\frac{1-m}{2}},\dots,x_0,\dots,x_{\frac{m-1}{2}} \right\}$ and $Z = \left\{ \frac{1-m}{2}\dots,0,\frac{m-1}{2} \right\}$ is an integer. The coefficients including a_0, a_1, etc. are obtained by linear least squares. The calculation formula is as:

$$x_r = a_0 + a_1 z_r + a_2 z_r^2 + \dots + a_k z_r^{k-1} \tag{6}$$

where x_r is the output of the fit, z_r is the data to be fitted, and a is the parameter to be solved.

There are m equations for x_r and the matrix X is denoted as:

$$X = Z \cdot A \tag{7}$$

where $X \in R^{m\times 1}$, $Z \in R^{m\times k}$ and $A \in R^{k\times 1}$. Determine $\underline{A}$ by the least squares fitting method.

The least squares method is then used to compute $\overline{A}$:

$$\overline{A} = (Z^T \cdot Z)^{-1} \cdot Z^T \cdot X \tag{8}$$

Finally, the smoothing coefficient matrix C_{ξ} for X is obtained as:

$$C_{\xi} = (Z^T \cdot Z)^{-1} \cdot Z^T \tag{9}$$

III. B. Point-by-point weighting method based on attention mechanism

Neural network parameter optimization uses stochastic gradient descent method, which means that only one training data point will be sampled for gradient calculation in each update. Therefore this paper tries to assign weights for each training sample point based on this idea. By introducing the attention mechanism, the attention point-by-point weight assignment method based on training error is finally proposed.

III. B. 1) Attention mechanisms

The introduction of attention mechanisms can make computers more focused on specific content. The attention mechanism is implemented by the attention function, which maps a "Query" and a set of "Key-Value" pairs into an output. These "Query", "Key", and "Value" are tensors, and the output can be expressed as the weighted sum of "Value", and the corresponding weights are calculated according to the "Query" and "Key" with a certain fitness function. The fitness function is used to measure the similarity between "Query" and "Key" to determine the importance of "Value".

The scale-scaled dot product attention mechanism is shown in Figure 1. Given a set of sequence data $\{x_1, x_2, \cdots, x_n, x_i \in \mathbb{R}^{1\times x_{in}}\}$, the query vector $\{q_1, q_2, \cdots, q_n, q_i \in \mathbb{R}^{1\times d_k}\}$, the key vector $\{k_1, k_2, \cdots, k_n, k_i \in \mathbb{R}^{1\times d_k}\}$ and the value vector $\{v_1, v_2, \cdots, v_n, v_i \in \mathbb{R}^{1\times d_v}\}$. For the current vector x_i for which attention weights need to be computed, its query vector q_i is dot-multiplied with all the key vectors $\{k_i\}_{i=1}^{n}$ to obtain the intermediate weight vectors $\{c_1, c_2, \cdots, c_n\}$, which are eventually transformed into a probability distribution $\{\alpha_1, \alpha_2, \cdots, \alpha_n\}$, and is used as the individual weights of the value vector. When the dimensions d_k of the query vector and the value vector are large, the softmax function is prone to bias, i.e., the larger c_i is, the larger the final weights are. To avoid this, a scaling factor $\frac{1}{\sqrt{d_k}}$ is usually introduced, which makes the attention called "scale-scaled dot product attention", and is calculated as follows.

$$\begin{aligned}(\alpha_1,\alpha_2,\cdots,\alpha_n) &= soft\max\left(\frac{1}{\sqrt{d_k}}(c_1,c_2,\cdots,c_n)\right)\\ &= soft\max\left(\frac{1}{\sqrt{d_k}}\left(q_i k_1^T, q_i k_2^T,\cdots,q_i k_n^T\right)\right)\end{aligned} \tag{10}$$

Dot product operations can be implemented in computers by highly optimized matrix multiplication, so the above query vectors, key vectors, and value vectors are rewritten in matrix form $Q \in \mathbb{R}^{n\times d_k}$, $K \in \mathbb{R}^{n\times d_k}$, $V \in \mathbb{R}^{n\times d_v}$, so that the attentional weights of all inputs can be computed simultaneously:

$$Attention(Q,K,V) = soft\max\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{11}$$

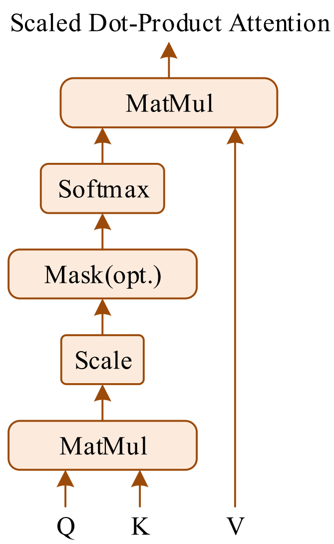

Figure 1: Scaled Dot-Product attention

III. B. 2) Error-Based Attention Point-by-Point Weight Assignment

The attention mechanism assigns weights to each training sample, which is very suitable for the idea of "point-by-point weighting", so the next question is how to calculate the attention weights. Considering that different partial differential equations have different training data, the following discussion is about the general form, and the loss function can be expressed as follows:

$$\begin{aligned}\mathrm{L}^* &= \frac{1}{N_b}\sum_{j=1}^{N_b}(w_b^j e_b^j)^2 + \frac{1}{N_i}\sum_{j=1}^{N_i}(w_i^j e_i^j)^2\\ &+\frac{1}{N_d}\sum_{j=1}^{N_d}(w_d^j e_d^j)^2 + \frac{1}{N_r}\sum_{j=1}^{N_r}(w_r^j r^j)^2\end{aligned} \tag{12}$$

Among them:

$$e_b^j = \mathrm{B}(x_b^j, t_b^j) - u(x_b^j, t_b^j;\theta) \tag{13}$$

$$e_i^j = \mathrm{I}(x_i^j) - u(x_i^j, 0;\theta) \tag{14}$$

$$e_d^j = u(x_d^j, t_d^j) - u(x_d^j, t_d^j; \theta) \tag{15}$$

$$r^j = \mathrm{N}\,[u(x_r^j, t_r^j; \theta), \gamma] - f(x_r^j, t_r^j) \tag{16}$$

Determining the attention weights requires three key pieces of information: Q, K, and V. From the above analysis, it is natural to assume that Q and K are generated from the coordinates of the data points, which are used to measure the spatial distribution of the samples, and V is generated from the training error or the residuals of the equations, i.e:

$$Q_s = X_s W_s^q \tag{17}$$

$$K_s = X_s W_s^k \tag{18}$$

$$V_s = \varepsilon_s W_s^v \tag{19}$$

where $s \in \{b, i, d, r\}$ is used to identify the sample point type, $W_s^q \in \mathbb{R}^{x_{in} \times d_k}$, $W_s^k \in \mathbb{R}^{x_{in} \times d_k}$, $W_s^v \in \mathbb{R}^{1 \times d_v}$ is the matrix of the linear transformation, and d_k, d_v are hyperparameters denoting the dimensions of the target space. $X_s = (x_s^1, x_s^2, \cdots, x_s^N)^T \in \mathbb{R}^{N_s \times x_{in}}$ is the training dataset, $\varepsilon_s = \left(e_s^1, e_s^2, \cdots, e_s^N\right)^T \in \mathbb{R}^{N_s \times 1}$ is the error during training. In fact there are various ways about the choice of E, the calculation of the above formula contains plus and minus signs, or it can be in absolute or squared form, such as $\varepsilon_s = \left(\left|e_s^1\right|, \left|e_s^2\right|, \cdots, \left|e_s^N\right|\right)^T \in \mathbb{R}^{N_s \times 1}$. Similarly, the attention weights are computed as follows:

$$Attention_s\left(Q_s, K_s, V_s\right) = soft\max\left(\frac{Q_s K_s^T}{\sqrt{d_k}}\right) c_s V_s \tag{20}$$

Since the parameters of the neural network are generally initialized based on a certain distribution, which can make the attention weights obtained in the early stages of training according to the above equation small, the balancing factor c_s can be multiplied for a specific term in order to amplify the significance of the initial margin condition. It should be noted that the scale factor $\frac{1}{\sqrt{d_k}}$ is not necessary when the dimension d_k is of small order of magnitude. In addition, since the result of the computation is a matrix of size $N_s \times d_v$, in this paper, the second dimension is averaged to make it into a weight vector of $N_s \times 1$.

The basic idea of the optimization process is that when the error of each item E_s is fixed at a certain moment, its weight should make the final loss function as large as possible to ensure that the attention mechanism can pay attention to the data points with higher errors. Therefore, in this paper, we introduce an adversarial training strategy to transform the original optimization problem into a two-layer optimization problem, which makes it possible to optimize the parameters of the neural network by minimizing the loss function, and at the same time maximize the loss function to optimize the parameters of the attention mechanism, i.e.

$$\begin{aligned} &\min_{\theta} \mathrm{L}^*\left(x; w^*, \theta\right) \\ &s.t.\ w^* = \arg\max_{w} \mathrm{L}^*\left(x; w, \theta\right) \end{aligned} \tag{21}$$

For such a very small and very large problem can be solved by alternating training, i.e., fixing the attention mechanism parameter w_1 for one-step optimization to get θ_1, then fixing θ_1 to get w_2, and so on.

III. C. Predictive Modeling

In order to solve the outliers in traffic flow and obtain better prediction accuracy, this paper proposes an attention-weighted SG-LSTM neural network model based on the above two techniques. An SG filter is used to reduce the input noise in $\{x_1, \ldots, x_t, \ldots, x_T\}$ (traffic flow before preprocessing). After noise reduction, the LSTM neural network encodes $\{\overline{x}_1, \ldots, \overline{x}_t, \ldots, \overline{x}_T\}$ (preprocessed traffic flow), using the encoder output c_T and the hidden state $d_{t'}$ of the decoder's previous step as the decoding of the current step t' as the Input. The output of the decoder is sent to the fully connected final layer for the final prediction result $\{\overline{y}_1,\ldots,\overline{y}_{t'},\ldots,\overline{y}_T\}$. The structure of the traffic flow prediction model is shown in Fig. 2.

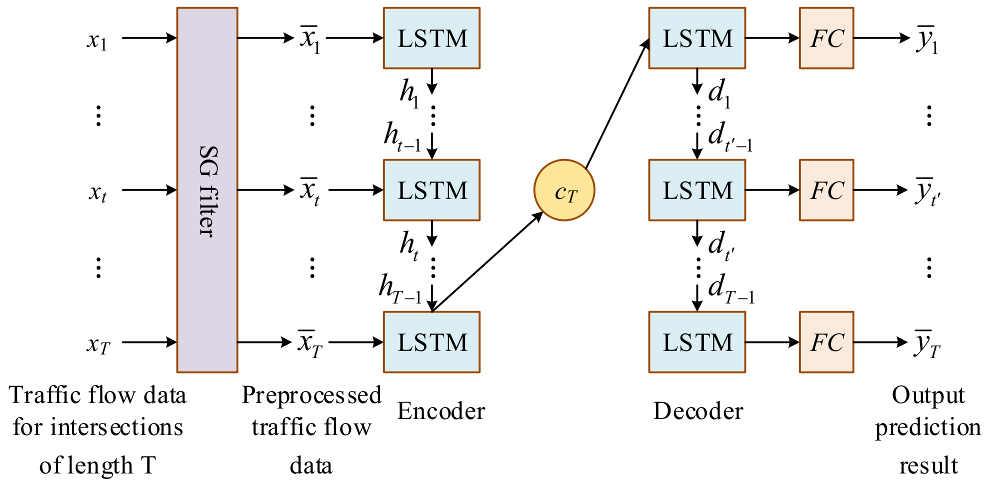

Figure 2: SG-LSTM neural network prediction model

The encoder network in the above figure is a sequentially transformed input $\{x_1,\ldots,x_t,\ldots,x_T\}$ LSTM Here,

$$h_t = f_1(h_{t-1}, x_t) \tag{22}$$

h_t is the hidden state of the encoder at t and f is a unit of the LSTM. c_T is the hidden state information obtained by the LSTM-based encoder.

The decoder is also an LSTM that produces the output by obtaining $\{\overline{y}_1,\ldots,\overline{y}_{t'},\ldots,\overline{y}_T\}$ for a given hidden state $d_{t'}$, and $d_{t'}$ conditional on c_T, with $d_{t'}$ representing the decoder's hidden state information at t'. Therefore, the formula for calculating $d_{t'}$ is as follows:

$$d_{t'} = f_2(d_{t'-1}, c_T) \tag{23}$$

The f_2 is another LSTM unit, and LSTM extracts complex features from time series. In multistep prediction, the output of the encoder c_T and the last step of the hidden state $d_{t'}$ from the decoder can improve the performance of multistep prediction. Fully connected layers can output them as predicted values:

$$\overline{y}_{t'} = w_k^{t'} \cdot d_{t'} \tag{24}$$

III. D. Analysis of the effectiveness of traffic flow forecasting

In this paper, we select the ETC gantry traffic flow in M City A to visualize the prediction results, which intuitively shows the prediction effect of the traffic flow prediction model built in this paper. Figure 3 shows the prediction results of the test samples. From the figure, it can be clearly found that the predicted value of traffic flow and the real value of traffic flow show a high degree of similarity, and the two traffic flow change curves do not show too much fluctuation. It shows that the prediction results of the neural network prediction model based on attention weighting constructed in this paper fit well with the real values.

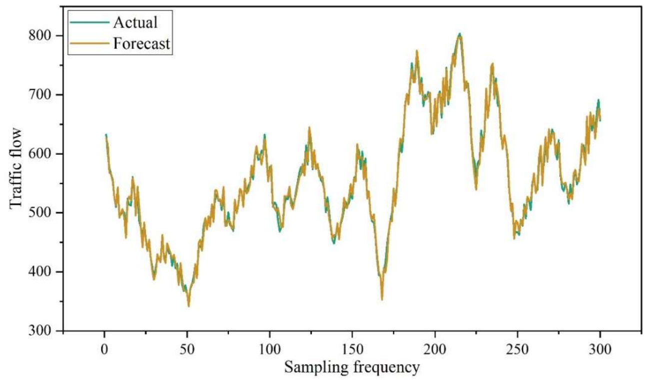

Figure 3: Test sample prediction results

The actual traffic network is complex and there is a lot of information contained in the ETC gantry traffic flow data. For this reason, four classical prediction models, LSTM, CNN, GCN and BP neural network, are also selected in this subsection to compare the prediction performance of ETC gantry traffic flow with the model of this paper, and the evaluation indexes are RMSE, MAE, R^2, and ACC as described above.

The results of the prediction performance comparison between the classical models and this paper's model are shown in Table 1. As can be seen from the table, the evaluation parameters MAE and RMSE of the LSTM model are 19.56 and 17.71 respectively in comparison with the performance of the classical prediction model.The evaluation parameters of the CNN model are 15.25 and 15.68 respectively, which are better than the LSTM model.The GCN model performs the best in the comparison algorithms, with the values of RMSE and MAE of 12.03 and 13.42 respectively.While the model of this paper RMSE, MAE values are reduced by 27.43%, 43.07% compared to GCN model. It can be concluded that the model in this paper is able to mine the information of spatio-temporal correlation of traffic flow, and the prediction error is lower. In addition, on the R^2 and ACC indexes, this paper's model also shows the best measured values, which are 0.862 and 0.926, respectively, and are 21.6 and 14.3 percentage points higher than the GCN model, respectively. The model in this paper is able to take into account the spatial dependence of the upstream and downstream of the target observation point and the neighboring observation points, which further improves the prediction accuracy of the model.

Table 1: The classic model compares the predictive results of this model

Model	RMSE	MAE	R^2	ACC
LSTM	19.56	17.71	0.549	0.717
CNN	15.25	15.68	0.434	0.694
GCN	12.03	13.42	0.646	0.783
BP	17.44	16.08	0.425	0.714
Ours	8.73	7.64	0.862	0.926

IV. Congestion warning model design and validation

IV. A. Early warning model design

In order to improve the responsiveness and accuracy of traffic management, a linkage study of traffic flow prediction and congestion warning is conducted. An attention-weighted SG-LSTM model is used for traffic flow prediction, and the data obtained from the prediction is subsequently input into a support vector machine (SVM) model for training, so as to implement an efficient and accurate warning. The warning model designed in this paper is shown in Fig. 4.

Traffic flow prediction: the traffic flow in a specified area is predicted using an attention-weighted long and short-term memory neural network.

Data input: the data output from the prediction model is passed to the early warning model through the linkage mechanism, which is accomplished through real-time data transmission and database sharing.

Early Warning Model: Using the received speed data combined with the early warning model, the current traffic state is evaluated to see if there is a potential risk of congestion.

Congestion Determination: The early warning model determines whether the current traffic state tends to be congested according to the predicted speed data and the analysis results of the early warning model.

Early warning generation: If the early warning model determines that there is a risk of congestion in the traffic state, the system generates the corresponding congestion warning information. This information can include the location of the congested flow, the expected arrival time.

Linkage Feedback: The congestion flow warning information is fed back to the speed prediction system through the linkage mechanism, and the congestion flow information is delivered to traffic managers, drivers and other related departments through real-time communication and visualization interface.

Real-time Adjustment: Traffic managers can adjust traffic flow control strategies in real-time according to the received congestion flow warning information, such as changing signal timing, adjusting lane flow allocation, etc., to avoid potential congestion.

Through such a speed prediction and congestion flow warning linkage process, the traffic management system is able to respond more intelligently to traffic changes, improve road utilization efficiency, and reduce the impact of congestion on the transportation system.

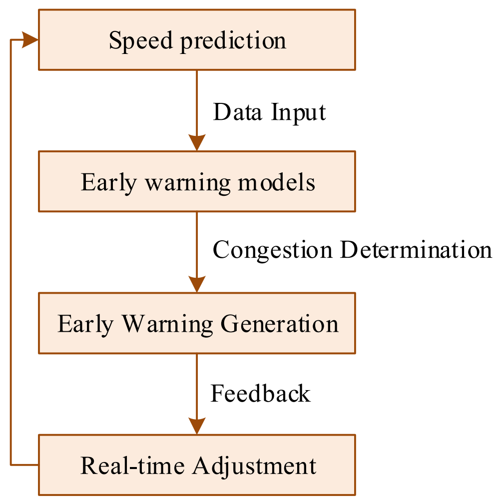

Figure 4: The correlation between the prediction and the early warning model

IV. B. Early warning model based on support vector machine

In the process of classifying the traffic congestion state, the SVM classifier is first learned and trained using the labeled learning sample set. By using the labeled learning sample set to learn the model parameters of SVM, the SVM classifier is equipped with the ability to classify, so that it can realize the classification of subsequent new input feature samples.

The sample set used by the SVM classifier is a new dataset obtained after the feature kernel function of the self-coding network, and let x be the new vector obtained after the kernel transformation of the feature vector i by the Sigmoid function. The labels are set for some of the learning samples, and this paper specifies that the labels are denoted by y_i, where $y_i \in \{1,2,3,4,5\}$, denote the five different traffic congestion states to be outputted by the SVM classifier, respectively. Therefore the learned training set of the SVM classifier is $L = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$.

The objective function of SVM is:

$$\begin{cases} \min \dfrac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i \\ s.t.\ y_i(w \cdot x_i + b) \geq 1 - \xi_i \\ \qquad \xi_i \geq 0, i = 1, 2, \dots, N \end{cases} \tag{25}$$

where C is the penalty factor. ξ_i is the slack variable. Calculation using the Lagrange function (i.e., the dyadic variable method) is as follows:

$$L(w,b,\alpha)=\frac{1}{2}\|w\|^2-\sum_{i=1}^{N}\alpha_i(y_i(w\cdot x_i+b)-1) \tag{26}$$

Finally the classification function is obtained as:

$$f(x)=sign\{w^*x+b\}=sign\left\{\sum_{i=1}^{N}y_i\cdot\alpha_i^*\cdot(x\cdot x_i)+b^*\right\} \tag{27}$$

where w^*, α_i^*, b^* are the parameters to determine the optimal hyperplane, and $(x\cdot x_i)$ is the dot product of two vectors.

Since the classification of traffic state is not a linear model, in order to solve the nonlinear differentiable problem, this paper introduces the mapping function and chooses the most commonly used RBF kernel function, as shown in the following equation:

$$\kappa\left(X_i,X_j\right)=\exp\left(-\frac{\left\|X_i-X_j\right\|^2}{2\sigma^2}\right) \tag{28}$$

Let: $\gamma=\frac{1}{2\sigma^2}$, then:

$$\kappa\left(X_i-X_j\right)=\exp\left(-\gamma\left\|X_i-X_j\right\|^2\right) \tag{29}$$

With the introduction of the RBF kernel function in the SVM model, the low-dimensional space computation in the original SVM model is mapped to a high-dimensional space to avoid the dimensional catastrophe problem. After introducing the RBF sum function, the final classification function used to detect instances is:

$$f(x)=sign\{w^*\phi(x)+b^*\}=sign\left\{\sum_{i=1}^{N}y_i\cdot\alpha_i^*\cdot\phi(x)+b^*\right\} \tag{30}$$

In SVM classifier, the selection of C and ξ_i will affect the experimental results to a certain extent, in order to make the model show better results, the values of C and ξ_i should be selected based on experience on the one hand, and on the other hand, it is necessary to continuously adjust to get the appropriate values of C and ξ_i by the accuracy of the output results.

IV. C. Application Cases of Traffic Congestion Warning

The regional traffic congestion warning model constructed in this paper is suitable for real-time traffic congestion warning. After obtaining the regional congestion warning speed and flow thresholds based on historical data, the regional speed values and flow values can be monitored in real time. When the speed and flow rate reach the thresholds and the speed decreases and the flow rate increases, the traffic congestion warning is triggered, and at this time the traffic control needs to be strengthened to alleviate the congestion.

The accuracy of the traffic congestion warning method proposed in this paper is further verified based on the historical traffic data of the ETC gantry in Zone A of M. The speed warning value of the ETC gantry in Zone A is 45km/h, and the flow warning value is 3,500 vehicles/10min, and Monday, August 12, 2024 is selected as the time of study. Analyze the time sequence of speed and flow rate change in the time period from 6:00 to 22:00 on Monday and mark the warning period. The speed and flow rate change curves are shown in Figure 5. The relevant data show that after reaching the warning speed threshold and flow rate threshold, the traffic enters into serious congestion. The Monday morning peak is more congested and the duration of congestion is longer, the warning time of the morning peak is between 8:00 and 9:00, while the warning time of the evening peak is generally after 18:00. The traffic warning law of the ETC gantry in Area A of M City is basically consistent with the traffic operation law, which verifies the accuracy and effectiveness of the method of this paper.

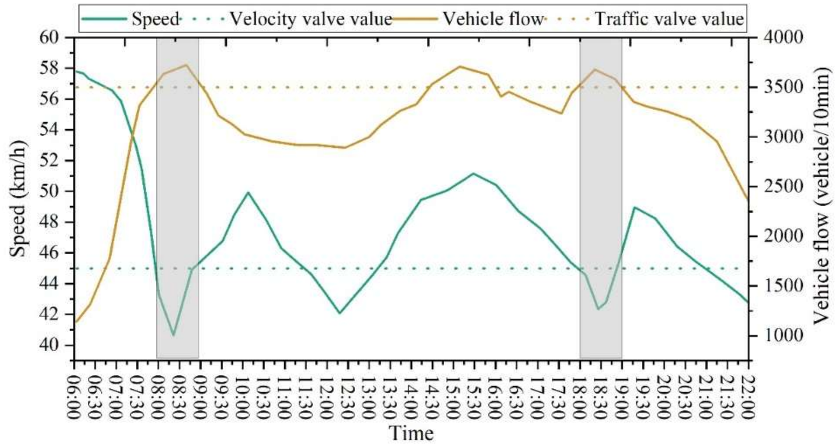

Figure 5: Velocity and flow curve

In order to further validate the effectiveness of the proposed congestion warning method, this study conducts a multi-day congestion warning experiment based on weekday data from August 12 to August 31, 2024 Figures 6 and 7 show the distributions of speed and flow for multi-day congestion warning, respectively. Both speed and flow data conform to normal distribution. In the figure, the warning speed range is about 45km/h, and the mean value of the fitted curve is 40.66km/h. The warning flow range is about 3500 (vehicles/10min), and the mean value is 2984 (vehicles/10min).

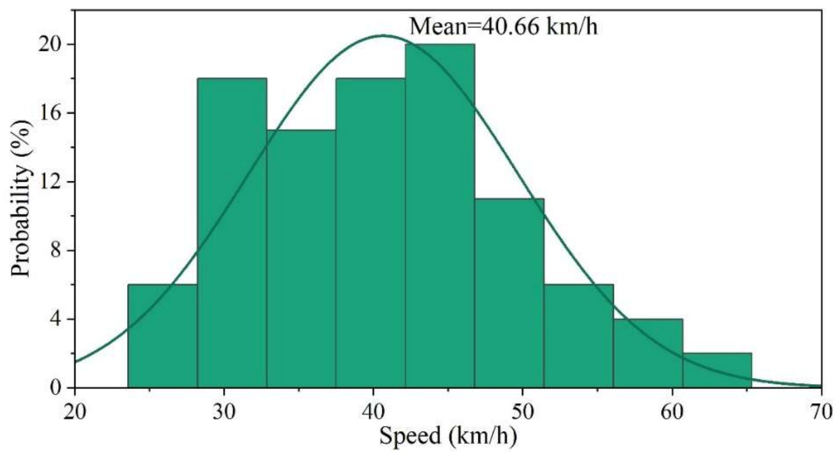

Figure 6: Multiple days of congestion warning velocity distribution

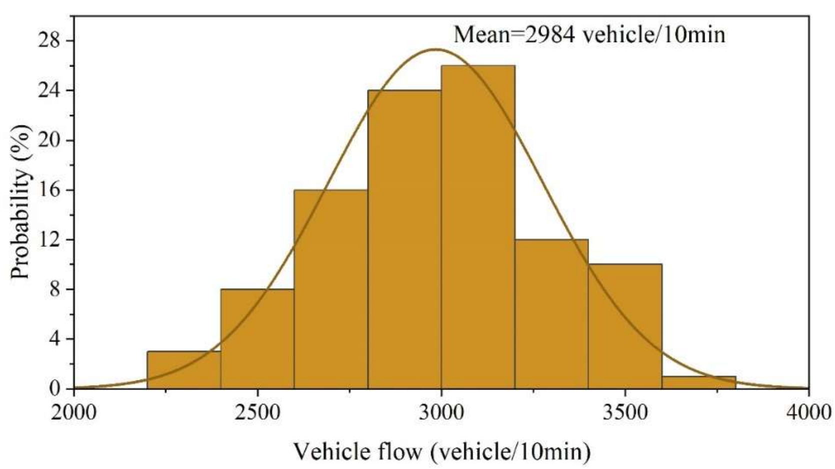

Figure 7: Multiple days of congestion warning traffic distribution

In order to verify the accuracy of the warning model proposed in this chapter, congestion warning analysis is conducted based on the weekday morning peak traffic data from August 12 to 31, 2024 It should be noted that only weekdays are selected for the analysis of the warning results, because the traffic conditions on weekends are quite different from those on weekdays. Table 2 shows the comparison between the model warning time and the actual congestion occurrence time. When the warning time is greater than or equal to the actual congestion time, the failure of the model is recorded as 0, and the success is 1. From the table, it can be seen that in the 15d warning results, only 1d warning is unsuccessful, and the accuracy of the warning can reach 93.33%. Through the verification of actual highway traffic flow data, it is found that the congestion warning model in this paper can provide technical support for ETC gantry control.

Table 2: Multiple days of congestion warning time results

Date	Actual traffic time	Predictive traffic time	results
2024.8.12	8:08	7:51	1
2024.8.13	8:03	7:56	1
2024.8.14	8:10	8:15	0
2024.8.15	8:08	7:56	1
2024.8.16	8:04	8:02	1
2024.8.19	8:02	8:01	1
2024.8.20	8:05	7:57	1
2024.8.21	8:06	7:53	1
2024.8.22	8:03	7:57	1
2024.8.23	8:03	7:59	1
2024.8.26	8:01	7:52	1
2024.8.27	8:08	7:54	1
2024.8.28	8:10	8:00	1
2024.8.29	8:07	7:55	1
2024.8.30	8:06	7:58	1

V. Conclusion

The SG-LSTM traffic flow prediction model based on attention weighting proposed in this paper significantly improves the accuracy of traffic flow prediction compared to the traditional LSTM, CNN, GCN and other models. By introducing Savitzky-Golay filter for data preprocessing, the interference of noise is reduced and the stability of traffic flow prediction is improved. In the experiment, the SG-LSTM model reduces 27.43% and 43.07% in RMSE and MAE, respectively. In addition, the congestion warning system combined with support vector machine effectively predicts the occurrence of traffic congestion through real-time data monitoring and prediction, and exhibits 93.33% warning accuracy in the test. This result indicates that the proposed model has high application value in practical traffic management and can provide accurate technical support for urban traffic congestion early warning.

References

[1] Nellore, K., & Hancke, G. P. (2016). A survey on urban traffic management system using wireless sensor networks. Sensors, 16(2), 157.

[2] Jain, V., Sharma, A., & Subramanian, L. (2012, March). Road traffic congestion in the developing world. In Proceedings of the 2nd ACM Symposium on Computing for Development (pp. 1-10).

[3] Javaid, S., Sufian, A., Pervaiz, S., & Tanveer, M. (2018, February). Smart traffic management system using Internet of Things. In 2018 20th international conference on advanced communication technology (ICACT) (pp. 393-398). IEEE.

[4] Rizwan, P., Suresh, K., & Babu, M. R. (2016, October). Real-time smart traffic management system for smart cities by using Internet of Things and big data. In 2016 international conference on emerging technological trends (ICETT) (pp. 1-7). IEEE.

[5] Das, A., Dash, P., & Mishra, B. K. (2017). An innovation model for smart traffic management system using internet of things (IoT). In Cognitive Computing for Big Data Systems Over IoT: Frameworks, Tools and Applications (pp. 355-370). Cham: Springer International Publishing.

[6] Sharif, A., Li, J. P., & Sharif, M. I. (2019). Internet of Things network cognition and traffic management system. Cluster Computing, 22(Suppl 6), 13209-13217.

[7] Guo, F., Zou, F., Luo, S., Chen, H., Yu, X., Zhang, C., & Liao, L. (2024). Positioning method of expressway ETC gantry by multi-source traffic data. IET Intelligent Transport Systems, 18(3), 540-554.

[8] Zhao, Y., Lu, W., Rui, Y., & Ran, B. (2023). Classification of the traffic status subcategory with ETC gantry data: an improved support tensor machine approach. Journal of Advanced Transportation, 2023(1), 2765937.

[9] Chen, H., Zou, F., Guo, F., Gu, Q., Yu, X., Luo, Y., & Xu, J. (2021, May). A ETC gantry information calibration method based on trajectory data of special transportation vehicles. In International Conference on Frontiers of Electronics, Information and Computation Technologies (pp. 1-7).

[10] Varotto, S. F., Jansen, R., Bijleveld, F., & van Nes, N. (2021). Driver speed compliance following automatic incident detection: Insights from a naturalistic driving study. Accident Analysis & Prevention, 150, 105939.
[11] Milić, D. C., Tolić, I. H., & Peko, M. (2020). Internet of Things (IoT) solutions in smart transportation management. Business Logistics in Modern Management.
[12] Pan, C., Guo, Q., Liu, J., & Min, X. (2024, February). Research on rapid warning model of expressway congestion abnormal events based on ETC gantry system. In International Conference on Smart Transportation and City Engineering (STCE 2023) (Vol. 13018, pp. 1263-1271). SPIE.
[13] Guo, F., Zou, F., Luo, S., Liao, L., Wu, J., Yu, X., & Zhang, C. (2022). The fast detection of abnormal ETC data based on an improved DTW algorithm. Electronics, 11(13), 1981.
[14] Ushakov, D., Dudukalov, E., Kozlova, E., & Shatila, K. (2022). The Internet of Things impact on smart public transportation. Transportation Research Procedia, 63, 2392-2400.
[15] Hussein, W. N., Kamarudin, L. M., Hussain, H. N., Zakaria, A., Ahmed, R. B., & Zahri, N. A. H. (2018, May). The prospect of internet of things and big data analytics in transportation system. In Journal of Physics: Conference Series (Vol. 1018, No. 1, p. 012013). IOP Publishing.
[16] Babar, M., & Arif, F. (2019). Real-time data processing scheme using big data analytics in internet of things based smart transportation environment. Journal of Ambient Intelligence and Humanized Computing, 10, 4167-4177.
[17] Lytras, M. D., Chui, K. T., & Liu, R. W. (2020). Moving towards intelligent transportation via Artificial Intelligence and Internet-of-Things. Sensors, 20(23), 6945.
[18] Awad, A. Y., & Mohan, S. (2019). Internet of things for a smart transportation system. International Journal of Interdisciplinary Telecommunications and Networking (IJITN), 11(1), 57-70.
[19] Xun Su,Yanmei Zhang,Yiyi Zhang,Jiefeng Liu,Min Xu & Pengfei Jia. (2025). An enhanced multilayer Res-Informer with Savitzky-Golay filter for predicting mixed CO and NOx emissions in gas turbines. Chemometrics and Intelligent Laboratory Systems,261,105379-105379.