

Research on Urban Road Traffic Flow Prediction Based on Hybrid CNN-LSTM Model

Jixiao Jiang^{1,a,*}, Anastasia Feofilova^{1,b}

¹Department of Transportation and Traffic Management, Don State Technical University, Rostov-on-Don, Russian Federation

a. 3396945068@qq.com, b. feofilowa@mail.ru

**corresponding author*

Abstract: Long-term traffic flow forecasting is crucial for intelligent transportation systems and also helps in the dynamic management of traffic flow. The goal of this research is to estimate future road traffic conditions using neural network models to reduce road congestion. Reducing prediction error is the most difficult task in traffic prediction. To predict traffic flow, this paper uses traffic data from Paris, France road dataset with normalized preprocessing of time series data. In this work, a novel CNN-LSTM hybrid model is built using MATLAB to predict traffic flow. The preprocessed data is used to train the CNN model, LSTM model and hybrid CNN-LSTM model. Finally, the predicted traffic flow data is evaluated using three evaluation indicators: coefficient of determination (R^2), mean absolute error (MAE) and root mean square error (RMSE). The results show that the proposed hybrid CNN-LSTM model can well capture and learn the long-term temporal characteristics of traffic flow and can predict traffic flow quickly and effectively.

Keywords: Intelligent Transportation System, Traffic Flow Prediction, CNN-LSTM

1. Introduction

Long-term traffic flow prediction refers to the prediction of traffic flow with a time span of more than 15 minutes, which is generally predicted on an hourly, monthly or yearly basis. It can predict the traffic flow over a longer time span in the future, which is of great significance to the design and planning of urban rail transit networks [1-3]. At present, prediction methods related to machine learning and neural networks are favored by researchers. In 1993, neural networks were first applied to the field of traffic flow prediction with their good nonlinear mapping ability, efficient data processing ability and strong anti-interference ability, and were well received by scholars at home and abroad [4]. Currently, neural network methods such as Artificial Neural Network (ANN), Feedforward Neural Network (FFNN), Recurrent Neural Network (RNN), Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) are frequently used to predict traffic flow [5,6]. Zhang et al. used CNN to extract the spatial features of trajectory traffic flow and predicted traffic flow [7]. Jang et al. extracted the temporal characteristics of trajectory traffic flow based on the memory characteristics of LSTM, which improved the accuracy of the prediction process [8]. Cheng et al. used a CNN-LSTM hybrid neural network model for multi-feature speed prediction at a spatial location and demonstrated that the prediction accuracy is correlated with the spatial correlation of traffic flow [9]. Wang et al. proposed a model that combines convolutional neural networks and

long short-term memory neural networks to efficiently identify long-term dependencies in traffic flow [10].

However, there is a limitation to existing deep learning-based traffic flow prediction efforts. For example, the simple neural network model architecture used in the LSTM model does not fully capture the complex features of the traffic flow, resulting in a slight error in the prediction performance. Therefore, in this paper, a novel hybrid model of Convolutional Neural Network (CNN) and Long Short-Term Memory Network (LSTM) is proposed in the framework of machine learning for predicting the traffic flow at 1 h intervals on the urban highways in Paris, France. By comparing the results of running different models, the effectiveness of the CNN-LSTM model is verified, thus providing a decision-making basis for traffic planning and traffic management of urban road networks.

2. Proposed Method

The CNN-LSTM model structure is shown in Figure 1, which mainly consists of a signal input layer, a CNN convolution layer, a pooling layer, a LSTM layer, and a classification output layer.

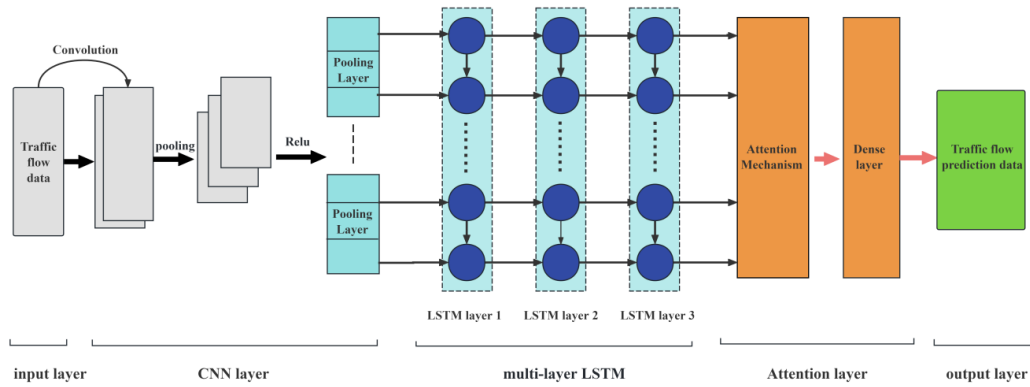


Figure 1: Overall structure of the model framework

2.1. CNN-LSTM model construction

Firstly, the Convolutional Neural Network (CNN) is modeled using MATLAB. The CNN structure is shown in Figure 2. The first layer inputs the image, performs convolution operation, obtains the feature map of the second layer, performs pooling operation on the feature map of the second layer, obtains the feature map of the third layer, and repeats the above operation to obtain the feature map of the fifth layer. Finally, these 5 feature maps are passed to the fully connected layer, and the output traffic flow data is obtained after deep learning.

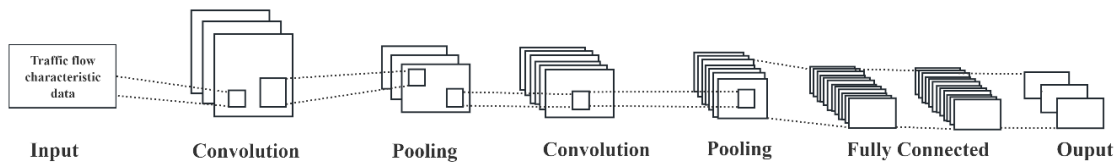


Figure 2: Convolutional Neural Network Model Structure

Since the input traffic flow feature map matrices are all square matrices, here let the input matrix size be w , the convolution kernel size be k , the step size be s , and the number of zero-complementing layers be p , then the formula for calculating the size of the resulting feature map after convolution is:

$$w' = \frac{(w + 2p - k)}{s} + l \quad (1)$$

After several layers of convolution and pooling operations, the obtained feature maps are expanded row by row, connected into vectors, and input into the fully connected network. The function of the convolution operation is as follows:

$$V = \text{conv2}(W, X, \text{"valid"}) + b, Y = \varphi(V) \quad (2)$$

Where, W is the matrix of different weights for each convolutional layer; X is represented as a matrix; conv2 is the function of the convolution operation in MATLAB; valid is the type of convolutional neural network; b is the bias; and $Y = \varphi(V)$ is the activation function. For the last fully connected layer set to layer L , the total error formula is as follows:

$$E = \frac{1}{2} \|d - y^L\|_2^2, \quad (3)$$

Where, y^L is the vector output form; d is the desired output.

The next step is to build a Long Short-Term Memory (LSTM) network model. Compared with traditional RNN (Recurrent Neural Network), LSTM introduces input gate, forget gate, and output gate, as shown in Figure 3. These mechanisms enable LSTM to better handle long-term dependencies in time series.

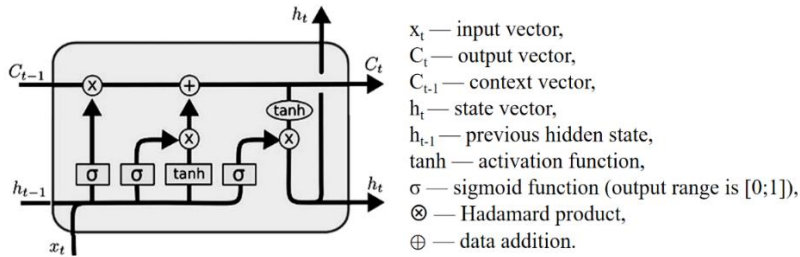


Figure 3: LSTM model basic architecture

In order to make the LSTM model have a good ability to identify traffic flow characteristics, it is used in time series prediction. The specific calculation formula is:

$$Y_{(t)} = w \times L(t) + (1 - w)G(t) \quad (4)$$

Where, $Y_{(t)}$ and $G(t)$ are the predicted values of the LSTM model, $L(t)$ is the prediction step, and w is the comprehensive weight.

2.2. Traffic flow data preprocessing

Traffic flow data is time series data, usually from sensors or GPS devices. In order to accurately predict traffic flow data, the data should be normalized first. The data used in this paper comes from the traffic flow dataset of urban roads in Paris, France (UTD19 // URL: <https://utd19.ethz.ch/index.html>). The traffic flow data after preprocessing is shown in Figure 4.

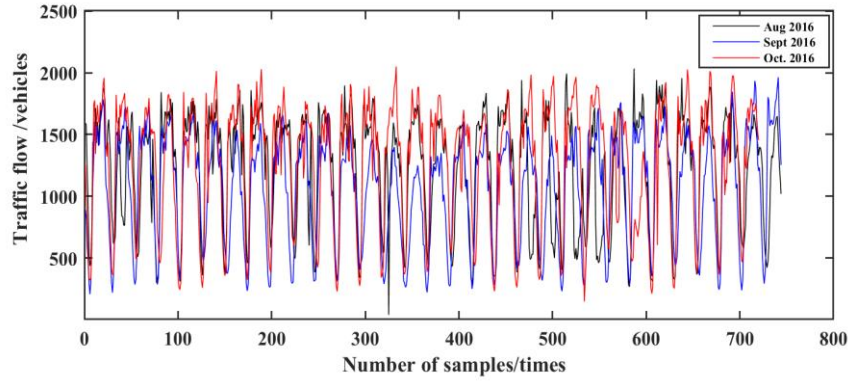


Figure 4: Normalized traffic flow data

2.3. Traffic flow prediction based on neural network model

The CNN-LSTM model structure established in this paper contains a one-dimensional convolution layer and an LSTM layer. The convolution is used for feature extraction and the LSTM is used for sequence modeling. Part of the code for running the model in MATLAB is shown in Figure 5.

```
%% Creating a CNN-LSTM Model
layers = [
    sequenceInputLayer(20) % Input Layer

    % First convolution layer combination
    convolution1dLayer(3, 16, 'Padding', 'same') % 1D convolution layer, small kernel
    batchNormalizationLayer % Batch normalization layer
    reluLayer % Relu activation layer

    % Second convolution layer combination
    convolution1dLayer(5, 32, 'Padding', 'same') % 1D convolution layer, medium kernel
    batchNormalizationLayer % Batch normalization layer
    reluLayer % Relu activation layer

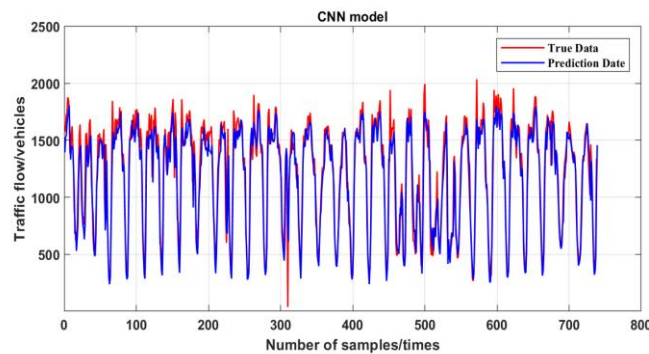
    % Third convolution layer combination
    convolution1dLayer(7, 64, 'Padding', 'same') % 1D convolution layer, large kernel
    batchNormalizationLayer % Batch normalization layer
    reluLayer % Relu activation layer

    lstmLayer(10, 'OutputMode', 'last') % LSTM layer
    reluLayer % Relu activation layer

    fullyConnectedLayer(1) % Fully connected layer
    regressionLayer]; % Regression layer
```

Figure 5: Part of the code of the CNN-LSTM model running in MATLAB

After deep learning of traffic flow data, the traffic flow data from September to November was input into three neural network models: CNN, LSTM, and CNN-LSTM. The predicted traffic flow data is shown in Figure 6.



(a)

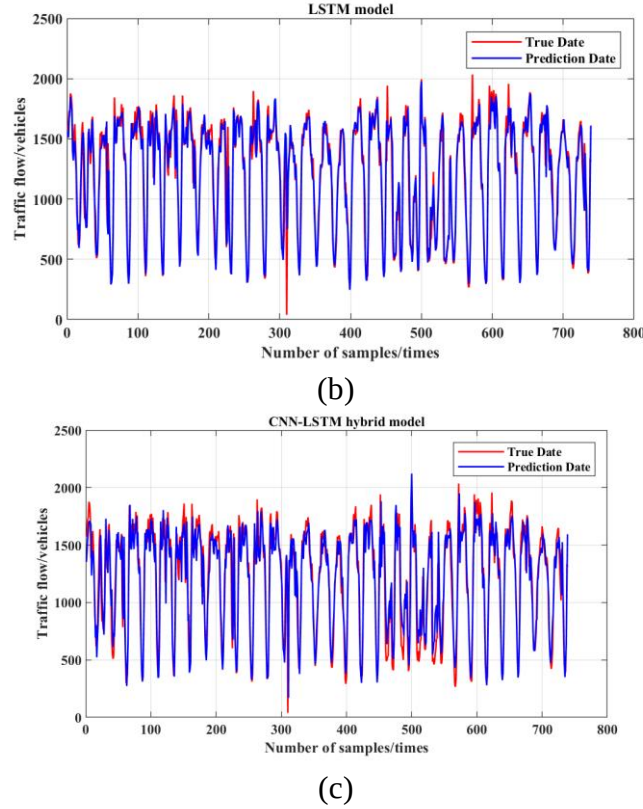


Figure 6: Traffic flow predicted by different neural network models: (a) CNN model, (b) LSTM model, (c) CNN-LSTM hybrid model

3. Experimental results analysis

To evaluate the effectiveness of the model in predicting traffic flow, the three neural network models must be run multiple times in MATLAB using traffic flow data from different months. The model is then evaluated using three key performance indicators: coefficient of determination (R^2), mean absolute error (MAE), and root mean square error (RMSE). R^2 evaluates the degree of model fit, and the larger the R^2 value, the better the model fit. MAE refers to the average error between the predicted traffic flow value and the actual value recorded at the detection point; the smaller the MAE, the higher the prediction accuracy of the proposed model. RMSE involves taking the square root of the average error calculated between the traffic flow predicted by the model and the actual traffic flow at the detection point; the smaller the RMSE, the higher the accuracy of the model. The formulas for the three evaluation indicators are as follows:

$$R^2 = 1 - \frac{\sigma_r^2}{\sigma^2} \quad (5)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (6)$$

$$RMSE = \left(\frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|^2 \right)^{\frac{1}{2}} \quad (7)$$

Where, σ_r^2 is the residual variance, σ^2 is the variance of the dependent variable y , y_i is the true value of traffic data at time point t , $\hat{y}_i$ is the predicted value of traffic speed at the same time point, and n denotes the number of captured traffic flows.

After multiple experiments to verify the effectiveness of the model, the three evaluation indicators of the predicted traffic flow are shown in Table 1.

Table 1: Evaluation metrics values output by different neural network models.

	Model type	September	October	November
R^2	CNN	0.8912	0.8823	0.9000
	LSTM	0.9158	0.8977	0.9033
	CNN-LSTM	0.9728	0.9803	0.9744
MAE	CNN	0.3168	0.3351	0.3463
	LSTM	0.2154	0.2298	0.2172
	CNN-LSTM	0.2391	0.2283	0.2279
RMSE	CNN	0.4426	0.4079	0.4473
	LSTM	0.3371	0.3773	0.4061
	CNN-LSTM	0.2677	0.2472	0.2515

4. Conclusion

The experimental results show that the R^2 values of the three models are not much different, all above 0.9. However, the R^2 value of the CNN-LSTM hybrid model is stable at around 0.97, so the model has the best fitting ability. The MAE values are almost all between 0.2-0.3, and the prediction effects of the three prediction models are good. The RMSE values are all less than 0.5, which proves that the predicted traffic flow is highly accurate. Therefore, the three neural network models can meet the needs of traffic flow prediction. However, the MAE and RMSE values of the CNN-LSTM hybrid model are always lower than those of the CNN and LSTM models, indicating that its data fitting accuracy is the highest. Therefore, the new CNN-LSTM model proposed in this paper can be competent for long-term traffic flow prediction tasks and can improve prediction efficiency.

References

- [1] Nicholas G. Polson, Vadim O. Sokolov, *Deep learning for short-term traffic flow prediction*, *Transportation Research Part C: Emerging Technologies*, Volume 79, 2017, Pages 1-17, ISSN 0968-090X, <https://doi.org/10.1016/j.trc.2017.02.024>.
- [2] W. Duan and Z. Gong, "Research and application of urban real-time traffic flow prediction based on STARIMA," *2022 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA)*, Dalian, China, 2022, pp. 1521-1525, doi: 10.1109/AEECA55500.2022.9919038.
- [3] H. Zou, H. Liu, T. Zhou, L. Jiashun and Y. Zhan, "Short-Term Traffic Flow Prediction using DTW-BiGRU Model," *2020 35th Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, Zhanjiang, China, 2020, pp. 557-562, doi: 10.1109/YAC51587.2020.9337579.
- [4] L. Yu, J. Zhao, Y. Gao and W. Lin, "Short-Term Traffic Flow Prediction Based On Deep Learning Network," *2019 International Conference on Robots & Intelligent System (ICRIS)*, Haikou, China, 2019, pp. 466-469, doi: 10.1109/ICRIS.2019.00122.
- [5] M. Muñoz-Organero and V. Corcoba-Magaña, "A Deep Graph Neural Network Approach for Assessing Origin Destination Traffic Flow Estimates Based on COVID-19 Data," in *IEEE Access*, vol. 12, pp. 119003-119014, 2024, doi: 10.1109/ACCESS.2024.3446619.
- [6] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). *Attention Is All You Need*. (Nips), 2017. arXiv preprint arXiv:1706.03762, 10, S0140525X16001837.
- [7] P. Zhang, K. Xie and G. Song, "A short-term freeway traffic flow prediction method based on road section traffic flow structure pattern," *2012 15th International IEEE Conference on Intelligent Transportation Systems*, Anchorage, AK, USA, 2012, pp. 534-539, doi: 10.1109/ITSC.2012.6338674.
- [8] Jang, H.-C.; Chen, C.-A. *Urban Traffic Flow Prediction Using LSTM and GRU*. *Eng. Proc.* 2023, 55, 86. <https://doi.org/10.3390/engproc2023055086>

- [9] Z. Cheng, J. Lu, H. Zhou, Y. Zhang and L. Zhang, "Short-Term Traffic Flow Prediction: An Integrated Method of Econometrics and Hybrid Deep Learning," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 5231-5244, June 2022, doi: 10.1109/TITS.2021.3052796.
- [10] J. Wang and C.O.N. Susanto, "Traffic Flow Prediction with Heterogenous Data Using a Hybrid CNN-LSTM Model," *Comput. Mater. Contin.*, vol. 76, no. 3, pp. 3097-3112, 2023. <https://doi.org/10.32604/cmc.2023.040914>