Differential Time-variant Traffic Flow Prediction Based on Deep Learning*

Wei Zhang, Fenghua Zhu, Yuanyuan Chen†, Xiao Wang, Gang Xiong and Fei-Yue Wang

Abstract—**The accuracy of traffic flow prediction significantly impacts the operation of Intelligent Transportation Systems (ITS). In this paper, we propose a Differential Time-variant (DT) Traffic Flow Prediction method, which can remarkably improve the accuracy and reduce the variance of traffic flow forecast based on deep learning models. To extract the temporal trend of the traffic flow at different locations, we apply data difference to preprocess the raw traffic data. This method can better eliminate the uncertainties of traffic flow series like volatility and anomaly. Then, time information is introduced in the form of One-Hot Encoding to effectively model the temporal patterns of traffic flow. Necessary analysis is presented to demonstrate the rationality. Three popular deep neural networks are applied to test our method, and experimental results on PeMS data sets indicate that it can make more accurate prediction compared with the same model.**

I. Introduction

In the past two decades, intelligent technologies have been widely used in the research and development of transportation systems, which used to be dominated by traditional industrial technology [1]–[4]. As an important part of transportation management and control, traffic flow prediction has also achieved a leap. Since the early 1980s, short-term traffic forecasting has been an integral part of most Intelligent Transportation Systems (ITS) and related researches, which is considered from a classical statistical perspective to all kinds of machine learning approaches [5].

Traffic flow prediction aims to estimate the target volume in the near future with some observed data [6]. With the emergence of deep neural network, data-driven methods based on deep learning models have achieved great progress in prediction task and obtained more superior performance compared with other machine learning approaches, which are our focus of this paper.

The traffic flow of a location at a certain time is affected by numerous factors that can not be fully taken into account. Thus, crucial variables that can make prediction system more accurate and stable are what we consider most, like historical traffic flow. Many data-driven researches about traffic flow prediction are based on the historical traffic flow, which is reasonable but not adequate. As traffic flow dynamics tend to be time-variant, the fluctuation tendency gradually changes over time. Thus, a smooth function might not be capable to reach accurate prediction for all periods. Although the intraday trend of traffic time series is common [7], like bimodal and unimodal types, many researchers neglect the influence of time information. Decided by individuals' daily life, similar time is more likely to correspond to similar situations. Therefore, time information might strongly impact the accuracy and stability of traffic flow forecast.

In this paper, we propose a One-Hot Encoding method to introduce time factors. This approach can achieve segmental prediction based on the time interval and improve the performance of deep neural networks.

There exist many uncertainties like volatility and anomaly in traffic time series. Drivers, vehicles and roads are all filled with intricate factors, and they are completely unpredictable. Thus, volatility is a common phenomenon and may greatly influence the learning process of the traffic flow prediction function. Another typical feature of traffic systems is anomaly, which can be contributed by accidents, detector failure, road construction etc. They are patterns in data that do not conform to a well-defined notion of normal behavior [8]. Anomaly does not frequently happen but is still common in traffic systems. It can impact greatly on traffic prediction for deep learning models. However, many data-driven prediction methods do not fully consider the training objective for prediction problems because of its rarity [9]. It is likely to be ignored by models as the proportion of prediction error is respectively small compared with the main patterns. Thus, anomaly can significantly weaken the stability of forecasting.

As data difference can improve the stationary property of time series, we utilize it to predict the traffic flow trend relative to last time. This can partly eliminate outliers of traffic flow series, as the focus is transformed from the actual volume to the temporary trend.

The contribution of this paper is threefold.

*This work was supported in part by the National Key R&Development Program of China under Grant 2018YFB1700202, NSFC U1811463, U1909204, 61773381, 61533019. Basic and applied basic research fund of Guangdong Province 2019B1515120030.

Wei Zhang is with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China. He is also with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, 100049, China.

Fenghua Zhu and Gang Xiong are with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China, and also with the Clouding Computing Center, Chinese Academy of Sciences, Dongguan 523808, Guangdong, China.

Yuanyuan Chen, Xiao Wang and Fei-Yue Wang are with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China.

†Corresponding author. E-mail: yuanyuan.chen@ia.ac.cn

- We demonstrate that introducing time information in the way of One-hot Encoding can improve the performance of deep learning models. This method takes time factors into account and can achieve even consistent forecast for some locations with strong time characteristics.
- Data difference is utilized to partly eliminate the influence of volatility and anomaly on deep learning models. It can effectively solve the problem of models' extremely inaccurate prediction for outliers to a considerable degree, and improve the performance.
- We combine the two methods and make experiments on PeMS data sets. Three popular deep learning models are tested to show the superior performance of our method, which are fully connected network (FC), long-short term memory network (LSTM) and convolutional neural network (CNN). Experimental results indicate that the models can make more accurate prediction with the proposed approach.

The rest of this paper is organized as follows. Section II briefly reviews the exsisting methods of traffic flow prediction. Section III introduces our methods comprehensively and structures of the latter two deep learning models in brief. Section IV presents the experimental results. Finally, Section V concludes this paper.

II. Literature Review

The traffic flow volume at time t can generally be stated as (1).

$$y_t = F(\vec{x}_{t-1}) + \epsilon_t \tag{1}$$

where y_t is the traffic flow volume of time t, $\vec{x}_{t-1} = [y_{t-N}, y_{t-N+1}, ..., y_{t-1}]^T$ denotes the historical traffic flow data. ϵ_t is served as an unpredictable part of prediction, which dynamically changes over time. It is usually regarded as white noise. The aim of traffic flow prediction is to acquire function F.

In the early stage, autoregressive integrated moving-average (ARIMA) [10] was used for traffic flow prediction, which may be the first data-driven method for practices. Soon afterwards, its variations, such as Kohonen-ARIMA [11], subset ARIMA [12] and so on, were applied to this area. Methods based on Kalman filtering model [13], k-nearest neighbor (kNN) algorithm [14], support vector regression (SVR) [15] and many other iterative algorithms were also proposed in the meanwhile.

Deep neural networks show great potential in traffic flow prediction as the technique has gained marvelous achievements in many other fields, which attracted the interests of researchers throughout the world. In the past few years, all kinds of models based on deep learning have been proposed. Huang *et al.* used a deep belief network (DBN) to learn effective features for traffic flow prediction in an unsupervised feature learning [16]. Lv *et al.* utilized stacked autoencoders (SAE) to extract the features of traffic flow and make short-term forecast [17]. As spatial features and temporal features are distinct for traffic flow, different structures were applied to extract the features separately then. LSTM, CNN and their combination like Conv-LSTM were used to acquire the dynamic characteristics of traffic flow [18]–[20]. Generative adversarial network (GAN) was also applied to traffic flow prediction since it was proposed in 2014 [21]. Since graph convolutional network (GCN) is capable of learning features of generalized topological graph structure, it was used to grasp the spatial correlation of different sensors [22]. Meanwhile, there are some other novel methods used to traffic flow prediction, like residual deconvolution [23] and fuzzy-based method [24].

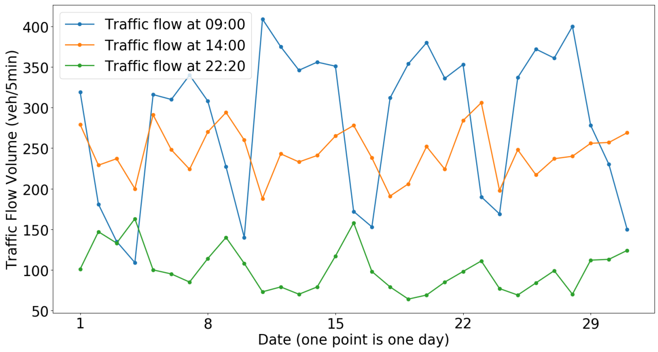


Fig. 1: Traffic flow volume at different time of different days. The data are collected by the sensor (No. 400132) from July 1, 2016 to July 31, 2016.

Although all kinds of deep learning based methods have been developed for traffic flow prediction, it is difficult to say that one method is the best for any situations. Thus, methods that are capable of boosting different networks to achieve more accurate and stable prediction are significantly important, which is exactly the focus of this paper.

III. Methodology

In this section, we will firstly introduce our methods on importing time factors and eliminating uncertainties. Some necessary analysis are presented to illustrate the rationality. Then the general structure based on deep learning models is proposed for utilization of our methods. Finally, the structure of LSTM and CNN used in our experiment will be briefly depicted.

A. Methods to Improve Models' Performance

1) Encoding Time with One-Hot: The intraday trend of traffic time series implicitly indicates that similar time often corresponds to similar traffic flow volume and similar traffic situation. As shown in Fig. 1, traffic flow of the same time presents similar characteristics, which are distinct at different time. This indicates that the actual flow volume has strong correlation with time.

Considering an one-to-one prediction task and a simplest fully connected network, the forecasting function can be depicted as (2).

$$\widehat{y}_t = F(\vec{x}_{t-1}) = f(\vec{w}^T \vec{x}_{t-1} + b) \tag{2}$$

where $\hat{y_t}$ denotes the predicted value, $\vec{w}$ and b are the learnable parameters. If one day's total amount of time is N_T, a one-hot shaped time vector $\vec{o}_t \in R^{N_T}$ can be defined as (3).

$$o_t^{(i)} = \begin{cases} 1, & \text{i = t} \\ 0, & \text{else} \end{cases} \quad i = 0, 1, ..., N_T - 1 \tag{3}$$

To take a freely changable time factor into the function, we change $\vec{x}_{t-1}$ into $[\vec{x}_{t-1}^T, o_t^T]^T$, which can get (4).

$$\begin{aligned} \hat{y_t} &= f([\vec{w}^T, \vec{w}_o^T] \begin{bmatrix} \vec{x}_{t-1} \\ \vec{o}_t \end{bmatrix} + b) \\ &= f(\vec{w}^T \vec{x}_{t-1} + \vec{w}_o^T \vec{o}_t + b) \\ &= f(\vec{w}^T \vec{x}_{t-1} + w_o^{(t)} + b) \\ &= f(\vec{w}^T \vec{x}_{t-1} + b_t) \end{aligned} \tag{4}$$

where $\vec{w}_o \in R^{N_T}$ is the newly added parameters corresponding to $\vec{o}_t$. As b_t can be completely different at different time, periodic forecast is achieved in the way of subsection mapping. It is not required to modify the structure of network again while achieving more superior performance. And this approach is also suitable for many-to-many prediction, as (5).

$$\hat{\vec{y_t}} = f[WG(X_{t-1}) + \vec{b_t}] \tag{5}$$

where X_{t-1} is the historical data of multiple sensors. $G(X_{t-1})$ is the features captured by a deep learning model.

The difference between (4) and (5) can be described as follows. For a specific deep learning model, we just concatenate the input data and $\vec{o}_t$ together for one-to-one prediction. However, the extracted features are concatenated with $\vec{o}_t$ for many-to-many predicition, which can achieve similar performance.

2) Eliminating Uncertainties with Data Difference: It is widely accepted that Moving-Average (MA) can effectively eliminate the stochastic volatility and improve the stationary property of traffic time series. MA assumes that the predicted value is mainly influenced by the historical accumulation of error terms. This is also suitable for deep learning models. Thus, we make the first-order difference of the historical traffic flow and replace the original data as the input of networks. It is a reasonable approach to eliminate the uncertainties of traffic time series to a certain degree, as follows.

$$y_{t+1} = F(\vec{x}_t) + \epsilon_{t+1} \tag{6}$$

$$\begin{aligned} F(\vec{x}_t) &\simeq F(\vec{x}_{t-1}) + \nabla^T F(\vec{x}_{t-1}) \Delta\vec{x}_t + \frac{1}{2} \Delta\vec{x}_t{}^T \nabla^2 F(\vec{x}_{t-1}) \Delta\vec{x}_t \\ &= F(\vec{x}_{t-1}) + \Psi(\Delta\vec{x}_t) \end{aligned} \tag{7}$$

where $\Delta x_t = \vec{x}_t - \vec{x}_{t-1}$ is the historical data after difference. Then we can get (8)

$$\begin{aligned} y_{t+1} &= F(\vec{x}_{t-1}) + \Psi(\Delta\vec{x}_t) + \epsilon_{t+1} \\ &= y_t + \Psi(\Delta\vec{x}_t) + \epsilon_{t+1} - \epsilon_t \end{aligned} \tag{8}$$

Where $\Delta\epsilon_{t+1} = \epsilon_{t+1} - \epsilon_t$ is closer to white noise compared with ϵ_{t+1}, which can be ignored under the training of big data. $\Psi(\Delta\vec{x}_t)$ can be aquired by training a deep learning model. This approach is effective to eliminate the uncertainties of traffic flow series, as it is not based on the actual traffic flow but the temporary trend. Take Fig. 2 as an example, the traffic flow volume of the last day is obviously higher than its daily situations. However, after difference, the three days' time series become very similar.

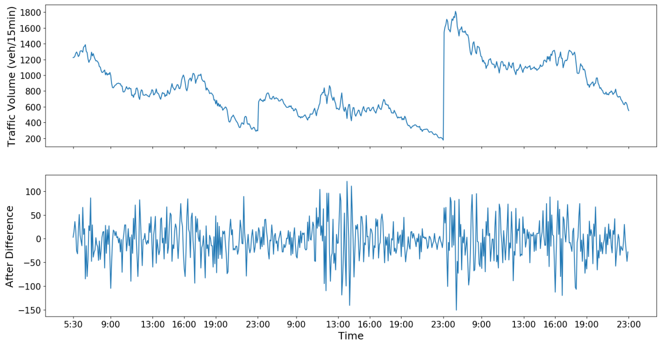


Fig. 2: Traffic flow volume before and after difference. The data are collected by the sensor (No. 400014) collected from July 12 to July 14.

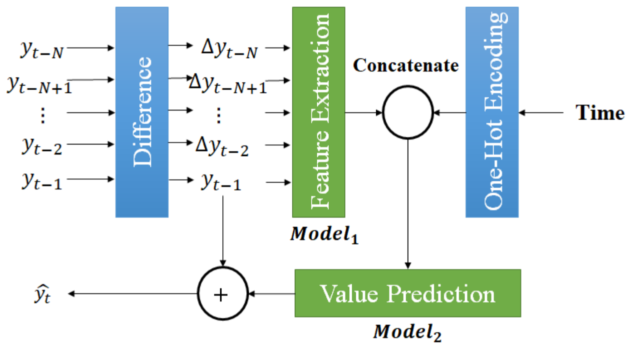


Fig. 3: The general structure of our method.

3) The General Structure: As the above two methods are complement with each other, we combine them together and find it effective to make more accurate prediction compared with the same model. The general structure of prediction network can be described as follows. After difference, the historical traffic flow data are fed into the first model to extract the time-invariant spatial-temporal features. The features will be combined with one-hot shaped time information then. The second network aims to obtain the predicted trend relative to last time with the concatenate features. Finally, add y_{t-1} to get y_t. See Fig. 3 for an illustration.

B. Deep Learning Based Models

1) Long short-term memory network (LSTM): LSTM is a special recurrent neural network (RNN) architecture, which is mainly used to solve problems with temporal dependencies. Unlike the traditional RNNs, LSTM were developed to tackle the exploding and vanishing gradient problems that can be encountered when training RNNs. LSTM is composed of many LSTM units. And each unit contains a cell, an input gate, a forget gate and an output gate. The computation process of one unit can be descirbed as follows.

$$
\begin{aligned}
i_t &= \sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}) \\
f_t &= \sigma(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}) \\
g_t &= tanh(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}) \\
o_t &= \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}) \\
c_t &= f_t * c_{t-1} + i_t * g_t \\
h_t &= o_t * tanh(c_t)
\end{aligned} \tag{9}
$$

where i_t, f_t, g_t, o_t are the input, forget, cell, and output gates, respectively. x_t is the input at time t, h_{t-1} denotes the hidden state of the layer at time $t-1$ and c_t is the cell state at time t. σ is the activation function, $*$ denotes the Hadamard product and others are learnable parameters.

To make traffic flow prediction, the historical volume of all sensors at each time step are formulated as a vector.

2) Convolutional neural network (CNN): The architecture of a CNN is inspired by the organization of cat's visual cortex, which uses different convolutional kernels to extract the features of image data. The 2D convolution process can be depicted as (9).

$$O_i^l = \sigma(b_i^l + \sum_k W_{i,k}^l * O_k^{l-1}) \tag{10}$$

where k and i are the channel indices of input and output, $*$ denotes the convolution operation, $W_{i,k}^l$ and b_i^l is the learnable parameters, and σ is the activation function.

To make traffic flow prediction, each time step's traffic flow volume of multiple sensors are arranged to a square grid. To make strong correlated sensors close, we rearrange the order by maximizing the sum of correlation coefficient between adjacent pixels, which can be described as (10) and has been proved effective in [19].

$$order^* = \arg\max_{order} \sum_{i=1}^{[\sqrt{N}]} \sum_{j=1}^{[\sqrt{N}]} \frac{1}{G_{i,j}} \sum_{g=1}^{G_{i,j}} Corr(P_{i,j}, P_{i,j}^g) \tag{11}$$

where N is the number of sensors, $G_{i,j}$ is the number of neighbor pixels at the corresponding coordinates, $P_{i,j}^g$ is the g^th neighbor pixel and $Corr$ denotes the operation of correlation coefficient.

IV. EXPERIMENTS

A. Dataset Description

The experimental traffic flow data are extracted from the Caltrans Performance Measurements Systems (PeMS) traffic database, which is aggregated in 5min interval. The particular data used in our experiments are chose from 100 detector stations located in district 4 from March 1, 2016 to August 31, 2016. Data in holidays are removed by KMeans, and then we get 128 days' traffic flow volume in this period. Simple average method is used to impute the missing data. As the volume are usually very small, data before 4:00 and after 23:00 are not considered in our experiments. Time number of one day is equal to the amount of its samples. That is to say, the serial number of a sample in one day is exactly its corresponding time. For all the models, we choose the data of the first five months as the training set and data of the last month for test. All experiments are performed on a desktop with an Intel Core i7-9700 CPU and an Nvidia Geforce GTX 1660 Ti (6G) Graphics Card.

B. Parameter Settings

In this paper, the size of time step is set to 5. 10% of the training data is selected as the validation set. Early stopping is used to decide the number of iterations. To determine the optimal parameters of LSTM and CNN on validation set, a grid search is conducted.

For LSTM, the hidden size is set as 300 from search space $\{200, 300, 400\}$. And we use a fully connected network to obtain the predicted value, whose number of each layer is $[300 + N_T, 300, 100]$.

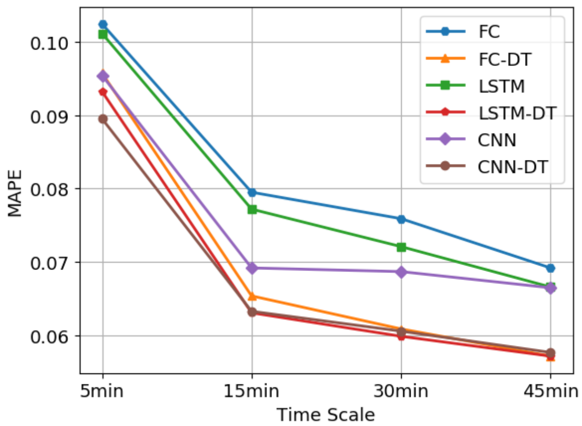
For CNN used in this paper, the number of hidden layers is set as 3 from space $\{2, 3, 4\}$. Two convolutional layers with 5×5 kernel are constructed to extract the spatial features of traffic flow data, and another one whose kernel size is 1×1 is used to grasp the temporal features. Similarly, a fully connected work is selected for value prediction.

C. Performance of our methods

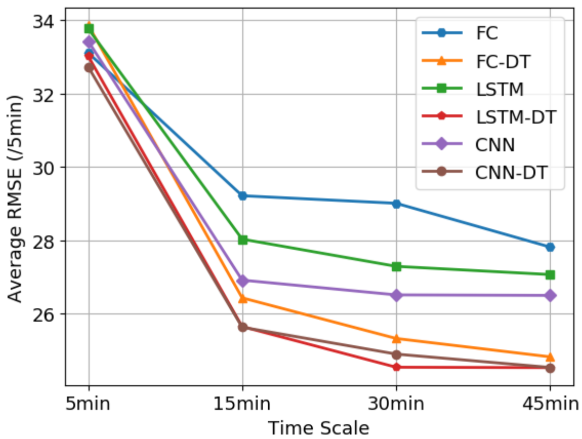
FC, LSTM and CNN are utilized to test the effectiveness of our method. And the results of some other approaches are also presented. We perform the 15-min traffic flow prediction task over the whole 100 detector stations, whose experimental results are shown in Table I. Here, With-T represents prediction with time, With-D represents prediction with data after difference and With-DT denotes traffic flow prediction with both methods. It can be clearly seen that our methods can significantly boost the prediction performance of the three deep neural networks. Thus, the approach is suitable for many deep learning methods. Take LSTM as an example, our methods improves the model's performance by 8.5%, 9.8%, 18.3% for RMSE, MAE and MAPE, respectively. Traffic flow prediction with time or data difference can improve

TABLE I: Performance of Multiple Prediction

Model		RMSE	MAE	MAPE(%)
ARIMA		112.28	74.05	9.45
SVR		86.98	60.16	7.40
FC	Ori	87.64	60.48	7.95
	With-T	83.45	57.12	7.51
	With-D	80.48	55.72	6.74
	With-DT	**79.29**	**54.57**	**6.54**
LSTM	Ori	84.09	58.15	7.72
	With-T	81.53	56.02	7.32
	With-D	79.07	54.77	6.65
	With-DT	**76.94**	**52.47**	**6.31**
CNN	Ori	80.74	56.48	6.66
	With-T	77.66	53.44	7.32
	With-D	79.34	54.99	6.58
	With-DT	**76.89**	**52.50**	**6.33**

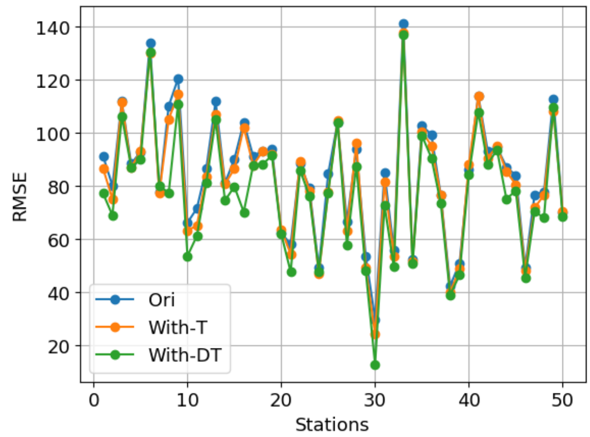


(a) MAPE at different time scales

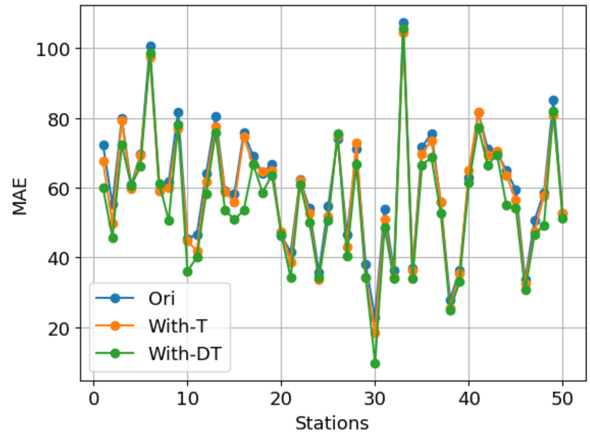


(b) Average RMSE at different time scales

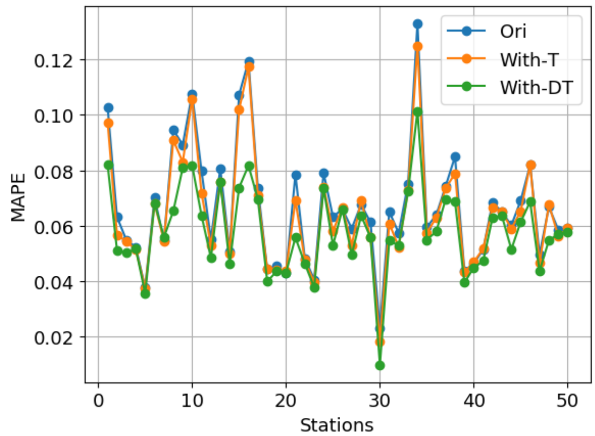
Fig. 4: Performance of models at different time scales.



(a) Performance comparison of RMSE



(b) Performance comparison of MAE



(c) Performance comparison of MAPE

Fig. 5: LSTM's performance of the first 50 stations from August 1 to August 31.

the models' accuracy of forecast, and the result will be better with both methods, which exactly conforms to our expectations.

Fig. 4 illustrates the performance of FC, LSTM, CNN with and without our methods at different time scales. Here, average RMSE denotes the value of RMSE per 5min. It can be easily found that the three deep learning models can achieve more superior performance at all time scales. And it is increasingly remarkable with the increase of time scale, which is reasonable as the time feature will be more stable and the uncertainties will be weaker.

Fig. 5 shows LSTM's prediction performance of the first 50 sensors. It indicates that models with our methods can make more accurate prediction, as the performance has been improved for most of the sensors.

To further analyze the effect of our methods, some variants of LSTM-DT are evaluated, including: (1) **DT2**: which compresses the dimension of time vector to the half, (2): **DT4**: which compresses the dimension of time vector to the quarter, (3): **Residual**: residual structure without data difference, (4): **Detrend**: which replaces data difference with detrending method. In this paper, we use weighted average to update the intraday trend, as (15).

$$Trend(k) = \alpha T(k) + (1-\alpha) Trend(k-1) \tag{12}$$

Where $Trend(k)$ is the intraday trend up to the k^th week, $T(k)$ is the traffic flow series of the k^th week and α is a trade-off parameter. The results are shown in Table II. We observe that with the increase of the time aggregation level, the model's performance will deteriorate. And data difference can further improve the prediction accuracy compared to the residual structure, which is better than some detrending methods.

TABLE II: Performance of LSTM-DT's variants over 15-min prediction

Variants	RMSE	MAE	MAPE(%)
DT2	77.43	52.99	6.50
DT4	77.85	53.42	6.55
Residual	77.87	53.32	6.63
Detrend	80.17	54.86	6.99
LSTM-DT	**76.94**	**52.47**	**6.31**

V. Conclusion

In this paper, we propose two methods to improve deep learning models' performance of traffic flow prediction. The first method embeds time information into deep neural networks in the form of One-Hot Encoding, which can extract the temporal trend of the traffic flow at different locations and achieve segmental prediction based on the time interval. The second method can partly eliminate the uncertainties of traffic time series in the form of data difference. We combine the two methods together and find it effective to improve the performance of deep learning models. Three popular deep neural networks are applied to test the availability of our method. Experimental results indicate that our method can significantly improve the accuracy of traffic flow prediction for the same model. It is worth mentioning that we just utilize a relatively simple approach, which can be further explored and derive more efficacious and rigorous methods.

References

[1] F. Wang, "Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, pp. 630–638, Sep. 2010.

[2] B. Ye, W. Wu, K. Ruan, L. Li, T. Chen, H. Gao, and Y. Chen, "A survey of model predictive control methods for traffic signal control," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, pp. 623–640, May 2019.

[3] T. Liu, B. Tian, Y. Ai, and F. Wang, "Parallel reinforcement learning-based energy efficiency improvement for a cyber-physical system," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, pp. 617–626, March 2020.

[4] P. M. Kebria, A. Khosravi, S. M. Salaken, and S. Nahavandi, "Deep imitation learning for autonomous vehicles based on convolutional neural networks," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, pp. 82–95, January 2020.

[5] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: Where we are and where we're going," *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 3 – 19, 2014. Special Issue on Short-term Traffic Flow Forecasting.

[6] Y. Chen, Y. Lv, X. Wang, and F. Wang, "Traffic flow prediction with parallel data," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 614–619, Nov 2018.

[7] C. Chen, Y. Wang, L. Li, J. Hu, and Z. Zhang, "The retrieval of intra-day trend and its influence on traffic prediction," *Transportation Research Part C: Emerging Technologies*, vol. 22, pp. 103 – 118, 2012.

[8] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, pp. 15:1–15:58, July 2009.

[9] Y. Lin, X. Dai, L. Li, and F. Wang, "Pattern sensitive prediction of traffic flow based on generative adversarial framework," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, pp. 2395–2400, June 2019.

[10] M. Ahmed and A. Cook, "Analysis of freeway traffic time series data by using box-jenkins techniques," *Transportation Research Record*, vol. 773, pp. 1–9, 01 1979.

[11] M. V. D. Voort, M. Dougherty, and S. Watson, "Combining kohonen maps with arima time series models to forecast traffic flow," *Transportation Research Part C: Emerging Technologies*, vol. 4, no. 5, pp. 307 – 318, 1996.

[12] S. Lee and D. B. Fambro, "Application of subset autoregressive integrated moving average model for short-term freeway traffic volume forecasting," *Transportation Research Record*, vol. 1678, no. 1, pp. 179–188, 1999.

[13] F. Yang, Z. Yin, H. Liu, and B. Ran, "Online recursive algorithm for short-term traffic prediction," *Transportation Research Record*, vol. 1879, pp. 1–8, 01 2004.

[14] H. Chang, Y. Lee, B. Yoon, and S. Baek, "Dynamic near-term traffic flow prediction: System-oriented approach based on past experiences," *Intelligent Transport Systems, IET*, vol. 6, pp. 292–305, 09 2012.

[15] M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, and L. D. Han, "Online-svr for short-term traffic flow prediction under typical and atypical traffic conditions," *Expert Systems with Applications*, vol. 36, no. 3, Part 2, pp. 6164 – 6173, 2009.

[16] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: Deep belief networks with multitask learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, pp. 2191–2201, Oct 2014.

[17] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, pp. 865–873, April 2015.

[18] R. Fu, Z. Zhang, and L. Li, "Using lstm and gru neural network methods for traffic flow prediction," in *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pp. 324–328, Nov 2016.

[19] X. Dai, R. Fu, E. Zhao, Z. Zhang, Y. Lin, F.-Y. Wang, and L. Li, "Deeptrend 2.0: A light-weighted multi-scale traffic prediction model using detrending," *Transportation Research Part C: Emerging Technologies*, vol. 103, pp. 142 – 157, 2019.

[20] Y. Liu, H. Zheng, X. Feng, and Z. Chen, "Short-term traffic flow prediction with conv-lstm," in *2017 9th International Conference on Wireless Communications and Signal Processing (WCSP)*, pp. 1–6, Oct 2017.

[21] L. Yisheng, Y. Chen, L. Li, and F.-Y. Wang, "Generative adversarial networks for parallel transportation systems," *IEEE Intelligent Transportation Systems Magazine*, vol. 10, pp. 4–10, 11 2017.

[22] Z. Li, G. Xiong, Y. Chen, Y. Lv, B. Hu, F. Zhu, and F. Wang, "A hybrid deep learning approach with gcn and lstm for traffic flow prediction*," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 1929–1933, Oct 2019.

[23] D. Zang, Y. Fang, Z. Wei, K. Tang, and J. Cheng, "Traffic flow data prediction using residual deconvolution based deep generative network," *IEEE Access*, vol. PP, pp. 1–1, 05 2019.

[24] J. An, L. Fu, M. Hu, W. Chen, and J. Zhan, "A novel fuzzy-based convolutional neural network method to traffic flow prediction with uncertain traffic accident information," *IEEE Access*, vol. PP, pp. 1–1, 01 2019.