Transportation Research Part C 90 (2018) 166–180

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

# A hybrid deep learning based traffic flow prediction method and its understanding

Yuankai Wu[a,b], Huachun Tan[a,b,*], Lingqiao Qin[c], Bin Ran[c], Zhuxi Jiang[b]

[a] Collaborative Innovation Center of Electric Vehicles in Beijing, Beijing Institute of Technology, No. 5 Yard, Zhong Guan Cun South Street, Haidian District, Beijing 100081, China

[b] School of Mechanical Engineering, Beijing Institute of Technology, No. 5 Yard, Zhong Guan Cun South Street, Haidian District, Beijing 100081, China

[c] TOPS Laboratory, Department of Civil and Environmental Engineering, University of Wisconsin-Madison, 1415 Engineering Drive, Madison, WI 53706, USA

ARTICLE INFO

*Keywords:*
Traffic flow prediction
Recurrent neural network
Convolutional neural network
Attention model
Deep neural network visualization

ABSTRACT

Deep neural networks (DNNs) have recently demonstrated the capability to predict traffic flow with big data. While existing DNN models can provide better performance than shallow models, it is still an open issue of making full use of spatial-temporal characteristics of the traffic flow to improve their performance. In addition, our understanding of them on traffic data remains limited. This paper proposes a DNN based traffic flow prediction model (DNN-BTF) to improve the prediction accuracy. The DNN-BTF model makes full use of weekly/daily periodicity and spatial-temporal characteristics of traffic flow. Inspired by recent work in machine learning, an attention based model was introduced that automatically learns to determine the importance of past traffic flow. The convolutional neural network was also used to mine the spatial features and the recurrent neural network to mine the temporal features of traffic flow. We also showed through visualization how DNN-BTF model understands traffic flow data and presents a challenge to conventional thinking about neural networks in the transportation field that neural networks is purely a "black-box" model. Data from open-access database PeMS was used to validate the proposed DNN-BTF model on a long-term horizon prediction task. Experimental results demonstrated that our method outperforms the state-of-the-art approaches.

## 1. Introduction

Accurate prediction of traffic flow is important in modern transportation systems. It is a booster for many applications which need reliable future traffic information. For example, the predicted traffic flow is an important reference for vehicle path planning (Yuan et al., 2011), it can help travelers make better route choice; Predicting where and when congestion will occur is greatly beneficial for transportation management, as practitioners would be able to allocate resources to the roads most at risk for congestion and ultimately reduce the traffic congestion; Predictive information can also help better use commercial vehicles. With its huge potential on numerous applications, traffic flow prediction has become a hot research topic over the last few decades.

In essence, traffic prediction is to make estimations of the future state based on experiences and knowledge extracted from related historical data. Therefore, techniques on data collection, transmission, storage, and mining have a huge impact on the prediction

[*] Corresponding author at: Collaborative Innovation Center of Electric Vehicles in Beijing, Beijing Institute of Technology, No. 5 Yard, Zhong Guan Cun South Street, Haidian District, Beijing100081, China.

*E-mail addresses:* Kaimaogege@gmail.com (Y. Wu), tanhc@bit.edu.cn (H. Tan), Lingqiao.qin@wisc.edu (L. Qin), bran@engr.wisc.edu (B. Ran), jzx42841934@163.com (Z. Jiang).

https://doi.org/10.1016/j.trc.2018.03.001
Received 2 April 2017; Received in revised form 9 January 2018; Accepted 1 March 2018
Available online 20 March 2018

methodologies (Ran et al., 2012). Recent development in those domains have introduced the notion of big data to transportation research (Zheng et al., 2016; Jin et al., 2013); big data brings us unprecedented opportunities for achieving prediction with utmost accuracy, and it also sets new demand to reform the traffic flow prediction modeling strategies.

Driven by the traffic data flood, a challenge arises: Can we make full use of latent knowledge hidden in big traffic data to forecast traffic flow? Prediction models with shallow-structured architectures have been used until now in most situations (Lv et al., 2015). These methods are simple and efficient on small sample data. However, they have limitations in dealing with large historical datasets and complex nonlinear functions (Bengio, 2009). Recently, deep learning-deep neural networks and combinations of neural networks have led to a series of breakthroughs for applications on complex and large datasets such as images, languages, and speeches (Krizhevsky et al., 2012; Hinton et al., 2012; Xu et al., 2015). Deep learning integrates inherent features extracted by multiple-layer architectures and classifier/regression in an end-to-end fashion. Recent evidences reveal that deep learning is a promising tool for traffic flow prediction. However, traffic flow is very complicated and contains plentiful characteristics in both space and time dimensions (Li et al., 2013; Tan et al., 2013b). It is not an easy task to develop a deep learning model that efficiently capture the characteristics of traffic flow.

Another more critical question is how to understand data-driven prediction methods using big data (Vlahogianni et al., 2014; Xu et al., 2016). In the transportation field, data-driven methods especially the neural network approach have long been criticized for its explain-ability. Neural network has long been regarded as a “black box” because it is difficult to understand due to the large number of neurons, complicated structures, and non-linear function (Karlaftis and Vlahogianni, 2011; Lippi et al., 2013; Zhang et al., 2014). Recently, there has been some improvement in the knowledge of how to visualize and understand deep neural network structure such as convolutional neural network (Zeiler and Fergus, 2014) and recurrent neural network (Karpathy et al., 2015). By taking a “surgical” approach in these “well-trained brains”, researchers not only obtain knowledge about how the neural networks work, but also a new understanding about the data such as languages and images the networks have read and seen. Since the big traffic flow data cannot be easily understood by traditional methods, we thus believe that it is interesting to explore how to extract new knowledge from traffic flow using a neural network that has been trained by big traffic data.

In order to address these issues in big data era, a novel traffic flow prediction method was proposed based on deep learning framework. Deep convolutional neural networks were utilized to mine the spatial features of traffic flow data. Meanwhile recurrent neural networks were employed to learn temporal features. In order to represent the multi-periodicity of traffic flow, traffic flow of both recent horizon and similar intervals of previous day and of previous week were fed into the prediction model. Considering that only part of historical traffic flow are highly related to the future, and the dependence is largely determined upon the traffic state of the transportation network, thus the speed data was used to train an “attention” model to dynamically attend to the salient part of recent traffic flow in this paper. The proposed model was evaluated on prediction of future traffic flow on a long future horizon (45 min traffic flow is recorded every 5 min) which was merely done before. In addition, we gave a “surgical operation” on the trained network. It shows that the deep neural network can learn certain knowledge from big traffic flow data and the knowledge can be extracted if processed properly.

## 2. Literature review

In this section we provide relevant background about previous work on traffic flow prediction, the combination of convolutional neural network and recurrent neural network, and recent development on neural network visualization.

### 2.1. Traffic flow prediction

Since the 1980s, researchers have begun to study short-term traffic flow prediction, which is deemed to be useful for the real-time traffic control (Okutani and Stephanedes, 1984). Because of their strength on handling non-linearity and universal approximability of unknown functions, neural network approaches have been frequently employed for traffic flow prediction from earliest researches to today. For example, Zheng et al. (2006) combined neural networks and bayesian inference to forecast future traffic flow; Jiang and Adeli (2005) developed a time-delay recurrent wavelet neural network model to forecast traffic flow and highlighted the importance of periodicity on long-term forecasting. Ma et al. (2015) applied long short-term memory network to forecast traffic speed, and demonstrated that long short-term memory structure can capture the long-term temporal dependence of traffic data. Besides the neural network approaches, there are numerous other different prediction methods such as the time series model (Kumar and Vanajakshi, 2015; Sun et al., 2003), the Kalman filter (Guo et al., 2014), the support vector regression (Wang and Shi, 2013), the k-nearest neighbor (Wu et al., 2015a), the gradient boosting tree regression (Zhang and Haghani, 2015), and the hybrid models (Lopez-Garcia et al., 2016; Allström et al., 2016). The detailed information on existing models can be also found in two recent papers (Vlahogianni et al., 2014; Lippi et al., 2013).

The deficiencies of aforementioned methods stems from the contradiction between big traffic data and their shallow structures (Lv et al., 2015). To overcome the problem, the deep architecture for short-term traffic flow forecasting has recently been studied in ITS. Huang et al. (2014) used a deep belief network to capture the spatial-temporal features of traffic flow and proposed a multi-task learning architecture to perform exit station flow and road flow forecasting. Similarly, Lv et al. (2015) proposed a stacked auto-encoder model based on traffic flow prediction method. Tan et al. (2016b) investigated different pre-training strategies of DNN for traffic flow prediction. Yang et al. (2016) developed a Stack Denoise Autoencoder method to learn hierarchical representation of urban traffic flow. Polson and Sokolov (2016) utilized deep learning to forecast traffic flows during special events. These methods all belong to fully-connect structure, there is no assumptions about the features in the fully-connected architecture, thus it is difficult for a fully-connected neural networks to capture representative features from the dataset with plentiful characteristics. To fully utilize the topological feature of urban crowd flow, Zhang et al. (2016a,b) proposed a novel methodology based on the convolutional neural network. Nevertheless, they simply treat time dimension of traffic flow as a channel of image data, which means the features of traffic flow on time dimension are ignored. In summary, each of these deep architectures have their own advantages and disadvantages. Therefore, how to build a deep architecture that fully utilize the characteristics of traffic flow is yet an opening and challenging question.

### 2.2. *Combination of convolutional and recurrent neural networks*

By exploiting the fundamental spatial properties of images and videos, the convolutional neural networks (CNNs) always achieve dominant performance on visual tasks. In addition, the recurrent neural networks (RNNs) can successfully characterize the temporal correlation and exhibits superior capability for time series tasks such as speech recognition and language translation. Thus several attempts have been made to combine the CNN and the RNN architectures especially for the realization of strong artificial intelligence. For example, several methods have made use of the CNNs and the RNNs for image/video description generation (Vinyals et al., 2015; Yao et al., 2015; Peris et al., 2016). Specifically, Vinyals et al. (2015) introduced the visual attention mechanism for caption generation. The combination of CNN and RNN have also been successfully applied to visual activity recognition (Donahue et al., 2015), sentiment analysis (Wang et al., 2016), video classification (Wu et al., 2015b), and 3D object classification (Socher et al., 2012).

Traffic flow data, which is similar to frequently studied data in machine learning area such as video and audio, has plentiful characteristics in both time and space domains. There are some obvious characteristics, for example: in space domain, traffic flow patterns on some location are more likely to have stronger dependencies at nearby locations (topological locality); in time domain, the traffic flow long time before even has a long-term impact on the future. Motivated by the successes of the CNNs and the RNNs and with the consideration of the spatial-temporal characteristics of traffic flow, a CNN and a RNN were combined as the basic frame of the proposed method in this study.

### 2.3. *Deep neural network visualization and understanding*

There have been some improvements in our knowledge of how the deep neural networks (DNNs) operate on visual and language tasks. The tools of knowledge extraction are very similar to surgical experiment on human body. The most naive tool is to simply visualize the activations produced on each layer of a trained DNN (Yosinski et al., 2015). Several studies proposed to reconstruct images from intermediate features by using gradient-based approaches (Mahendran and Vedaldi, 2015; Simonyan et al., 2013) or another neural network (Zeiler and Fergus, 2014). It is found that the DNN progressively produces more invariant and abstract notion of the inputs. Another approach is to try to interpret the function computed by each individual neuron or their group (Zintgraf et al., 2016). For example, Karpathy et al. (2015) found that the neurons of a deeper layer are more likely to maintain long-term memory in the stacked RNN structure. Some intriguing studies used simulated data to stimulate the trained-DNN, Szegedy et al. (2013) revealed that a DNN can label a changed image imperceptible to humans to something else.

In summary, the majority of visualization and understanding work are made in visual tasks. How the DNN operates on other datasets such as traffic flow and weather data remains an open research area and is very worth for deeper understanding of DNN. In traffic flow prediction, the knowledge about prediction methods is gained from a result-centric approach (simply compare results of different methods), which is sensitive to test data and model parameters. With limited understanding of how and why they work, the development of better prediction models is reduced to blindly experiments and parameter-tuning. Thereby, it is urgent to study how to understand data-driven traffic flow prediction models.

## 3. Prediction model

In this section, we describe our deep learning based traffic flow prediction method (see Fig. 1) for the graphical illustration of the proposed model, which is largely inspired by the neural network zoo, which is a neural network drawing method.[1] Considering the periodicity of traffic flow, the traffic flow of similar horizon of previous day and previous week were input to the neural network with the near-term traffic flow data. The attention model focuses on the fundamental issue of traffic flow propagation, that is to select near-term data which is highly correlated to future traffic flow. The attention learned by fully-connected network from near-term traffic speed allows considering the weights of near-term traffic flow and reveals the dynamics between traffic speed and traffic flow. The spatial features of traffic flow are learned by the CNN and the temporal features of traffic flow are captured by a gated RNN, the prediction is fulfilled by concat all features together and input them into a regression layer. The whole network is trained end-to-end. We will describe each part in the following subsection.

### 3.1. *Attention model*

The attention model is aimed to determine the scores that how strong the input flow of past spatial-temporal position $s_t$ correlates to the future traffic flow. Since it is natural to assume the traffic flow exhibits stronger correlations between adjacent time points,

[1] http://www.asimovinstitute.org/neural-network-zoo/.

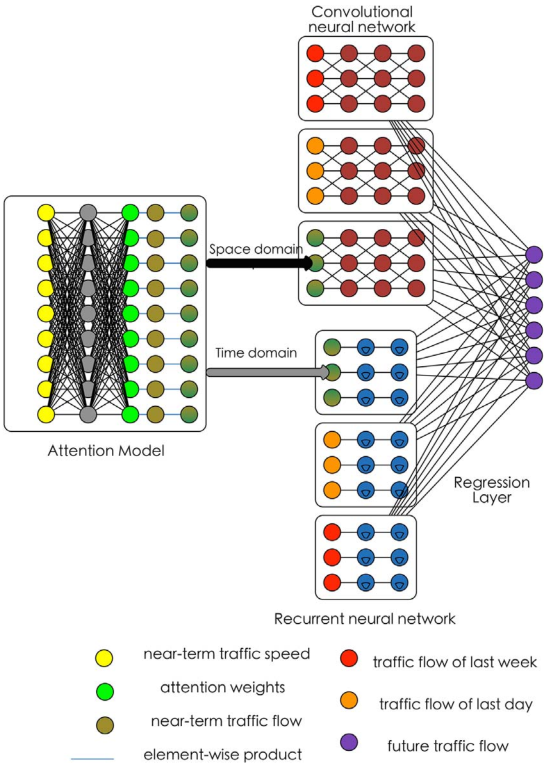


**Fig. 1.** A graphical illustration of proposed model.

scoring more heavily the recent past to the more distant past becomes a reasonable choice (Min and Wynter, 2011; Habtemichael and Cetin, 2016). Indeed, in reality the temporal correlation of traffic flow is affected by numerous external factors such as weather, incidents, and time of the day. Therefore, the scores are very difficult to determine, and in a broad spatial-temporal level, which is more complicated. Even distant traffic flow at one point maybe more important than recent traffic flow at any other point for forecasting the future traffic flow. Simply scoring the weights based on the recentness or other rough prior knowledge is insufficient.

Different from existing approaches, a fully-connected neural network was designed to learn the weights rather than directly use a rule-based strategy to determine this relationship. Since it is well known that the traffic speed is a key factor on how traffic flow propagate in future, therefore our approach makes use of the speed data to learn these weights. Assume we need to forecast traffic flow of $p$ locations $\{s_i\}_{i=1}^{p}$ in $(t,t+1,...,t+h)$, in which $h$ is the prediction horizon, the historical traffic flow data of these $p$ locations $\{s_i\}_{i=1}^{p}$ during $(t-n,t-n+1,...,t-1)$ are used as the inputs for generating predictions for the next time horizon $(t,t+1,...,t+h)$. Put together the historical data, a near-term traffic flow matrix is obtained:

$$\mathbf{S^f} = \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_p \end{bmatrix} = \begin{bmatrix} s_1(t-n) & s_1(t-n+1) & \cdots & s_1(t-1) \\ s_2(t-n) & s_2(t-n+1) & \cdots & s_2(t-1) \\ \vdots & \vdots & & \vdots \\ s_p(t-n) & s_p(t-n+1) & \cdots & s_p(t-1). \end{bmatrix}. \tag{1}$$

The goal of the attention model is to use a speed matrix $\mathbf{S^s}$ of these space-time points to learn an attention weight matrix $\mathbf{A}$ with the same size of $\mathbf{S^f}$. Each element in attention matrix $\mathbf{A}$ can be interpreted as the probability that the traffic flow information of that space-time point causes the future traffic flow, or as the relative importance of this space-time point for future forecasting. For simplicity, we directly use a fully-connected neural network with one hidden layer to learn this attention matrix. The activation function between outputs and hidden layer is a sigmoid function. The reason for using the sigmoid function is that it limits the outputs between 0 and 1.

$$\mathbf{A} = \mathfrak{F}_{\mathfrak{a}}(\mathfrak{f}(S^s)) = sigmoid(W\mathfrak{f}(s^s) + b), \tag{2}$$

where $\mathfrak{F}_{\mathfrak{a}}$ is the attention learning network, $s^s$ is the vectorization of speed matrix $\mathbf{S^s}$ with the same size of $\mathbf{S^f}$,$\mathfrak{f}$ is the projection between the inputs and the hidden neurons. Here we use a fully-connected network, but other different structures can be also used. Then the traffic flow matrix $\mathbf{S^f}$ is point-wise multiplied with the attention matrix $\mathbf{A}$ to obtain a weighted traffic flow matrix $\mathbf{S^A}$ for deeper learning procedures.

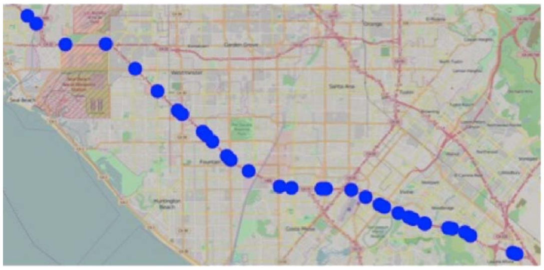


**Fig. 2.** Traffic flow locations studied in this paper.

### 3.2. Spatial feature mining

In what follows, we are interested in mining the spatial feature within traffic flow data, for which we choose the CNN structure. The traffic flow usually has a stronger correlation on nearby locations (Yang et al., 2015; Ermagun et al., 2017). As CNN is a very powerful tool for handling image/video data with the locality structure, CNN structure has been naturally utilized in some works (Zhang et al., 2016a; Zhang et al., 2016b). In this study,for the traffic flow from upstream $s_1$ to downstream $s_p$ on a freeway corridor as shown in Fig. 2, the conventional 1D CNN is exploited to capture the spatial features. In order to mine the spatial feature, the 1D convolution is performed on vectors $T_q = [s_1^a(t-n+q),s_2^a(t-n+q),\cdots,s_p^a(t-n+q)]^T (0 \geqslant q \leqslant n-1)$ of matrix $\mathbf{S^A}$ from upstream to downstream, where the $k$-th feature map is obtained as follows:

$$h_q^k = o_c(w_q^k * T_q^k + b_q^k), \tag{3}$$

where the $w_q^k$ is the weight vector, the $b_q^k$ is the bias, the $o_c$ denotes a nonlinear activation, and the $*$ denotes the convolution.

We use a full convolutional network in this paper; the pooling layers are not employed in our model. Because it is evident that a full convolutional net achieves better performance on small images recognition (Springenberg et al., 2014), and the space dimension of traffic data on this task is limited. In general, the deeper the network is, the stronger the representation capability of the network is. Nevertheless, deeper networks need more training data and are easier to overfit. For balance, the depth $k$ of the 1D CNN is set as 3. For a deep convolutional neural network, the Rectified linear activation units (ReLU) are always used as its nonlinear activation $o_c$ because the ReLU can solve the problem of “exploding/vanishing gradient” and can make the deep networks converge fast (Nair and Hinton, 2010). Considering the nonlinear relationship of the traffic flow data, the S-shaped Rectified Linear Activation Units (SReLU) was chosen in the proposed DNN-BTF model, because of its strong ability on learning non-linear transformation (Jin et al., 2015). The SReLU is formulated as:

$$o_c^s(x_i) = \begin{cases} t_i^r + a_i^r(x_i - t_i^r), x_i \geqslant t_i^r \\ x_i, t_i^l < x_i < t_i^r \\ t_i^l + a_i^l(x_i - t_i^l), x_i \leqslant t_i^l, \end{cases} \tag{4}$$

where the $t_i^r$, the $t_i^r$, the $a_i^r$, and the $a_i^l$ are parameters that the network need to learn.

There are many more complex transportation networks than the one given in Fig. 2, such as a transportation network of a big city. In a more complex transportation network, the conventional 1D CNN may not be able to fully characterize the spatial features. However the traffic flow in any transportation network always has some graph structures (Shahsavari and Abbeel, 2015). Thus the CNN on graph-structured data proposed by Henaff et al. (2015) might be an alternative for such complex transportation networks, which will be a valuable future direction.

### 3.3. Temporal feature mining

The temporal features of traffic flow is significantly different from its spatial features. Traffic flow exhibit stronger correlation in a short time period due to the dynamic nature of transportation system. Moreover, the long-term temporal dependency also exists within traffic flow data. An example of this dependency is the traffic flow upstream would take a long time to travel to downstream during extreme congestion. Hence the long-term dependency should be taken into account when modeling temporal features. Recently, the long short-term memory (LSTM) network has been introduced to model temporal features of traffic flow (Ma et al., 2015; Duan et al., 2016). The advantages of the long short-term memory is that it uses gated neurons to capture both the short-term and the long-term memories within traffic flow and to avoid the gradient vanishing/exploding problem.

A simpler RNN model-Gated recurrent neural network (GRU) is used in this study since it contains fewer neurons than that of the LSTM. It is also reported that GRU achieves equal or better performance than the LSTM (Cho et al., 2014). For the generations of short-term temporal features, the inputs of the GRU is denoted as $T = (T_0,T_1,...,T_{n-1})$ where $T_q = [s_1(t-n+q),s_2(t-n+q),...,s_p(t-n+q)]^T$ in $\mathbf{S^A}$, and the $k$-th layer output temporal features in each historical time point is denoted as $H^k = (H_0^k,H_1^k,...,H_{n-1}^k),n$ is the time window size. The generated temporal features are iteratively calculated by the following equations:

$$Z_q = \sigma_g(W_z T_q + U_z H_{q-1} + b_z), \tag{5}$$

$$R_q = \sigma_g(W_r T_q + U_r H_{q-1} + b_r), \tag{6}$$

$$H_q = (1-Z_q) \odot H_{q-1} + Z_q \odot \sigma_h(W_h T_q + U_h(R_q \odot H_{q-1}) + b_h), \tag{7}$$

where $\odot$ represents the point-wise product of two vectors, and $\sigma(.)$ and $\sigma_h(.)$ are activation functions. Usually, $\sigma(.)$ is set to be in the range of [0,1] to control the information flow through time. $\sigma_h(.)$ is often set to be a centered activation function. $R_q$ is the reset gate and $Z_q$ is the update gate at $q$ time point. It is obvious the network tends to learn long-term memories if most $Z$ are near zero and most of $R$ are near one, in reverse it tends to learn short-term memories. Because it is suggested that shallow single-layer GRU and LSTM only capture short-term memories (Karpathy et al., 2015), we stack two GRUs to extract the long-term memories and short-term memories of traffic flow in contrast to previous single layer model.

### 3.4. Periodic features

Commuters routinely go to work in the morning and go home in the evening on weekdays and travels differently on weekends, which is why a strong periodicity is observed within traffic flow. The periodicity of traffic flow have been identified as a major contributing factor for traffic flow forecasting. A desirable model that successfully characterizes the periodicity can accurately forecast future traffic flow (Tan et al., 2013a, 2016a; Wu et al., 2017). To model the daily and the weekly periodicity of traffic flow, the inputs of periodicity at time $t$ are given as follows:

$$\mathbf{S^d} = \begin{bmatrix} s_1(t^d-n^d) & s_1(t^d-n^d+1) & \cdots & s_1(t^d+h+n^d) \\ s_2(t^d-n^d) & s_2(t^d-n^d+1) & \cdots & s_2(t^d+h+n^d) \\ \vdots & \vdots & & \vdots \\ s_p(t^d-n^d) & s_p(t^d-n^d+1) & \cdots & s_p(t^d+h+n^d), \end{bmatrix}. \tag{8}$$

$$\mathbf{S^w} = \begin{bmatrix} s_1(t^w-n^w) & s_1(t^w-n^w+1) & \cdots & s_1(t^w+h+n^w) \\ s_2(t^w-n^w) & s_2(t^w-n^w+1) & \cdots & s_2(t^w+h+n^w) \\ \vdots & \vdots & & \vdots \\ s_p(t^w-n^w) & s_p(t^w-n^w+1) & \cdots & s_p(t^w+h+n^w), \end{bmatrix}. \tag{9}$$

where $t^d$ and $t^w$ denote the same time point of the first prediction time point $t$ in the previous day and in the previous week, respectively; $n^d$ and $n^w$ denote the time lags of daily periodicity and weekly periodicity, respectively; and $h$ is the prediction horizon.

The same CNN and GRU structures on near-term data are used to mine the spatial-temporal features of inputs of daily periodicity and weekly periodicity respectively. Then all the features are concated together and input into a regression layer to perform forecasting future traffic flow in $(t,t+1,...,t+h)$. Rectified Linear Activation Units (ReLU) (Eq. (10)) is used as the activation of the regression layer since it guarantees the non-negativity of traffic flow data and is liable to converge.

$$f(x) = max(0,x). \tag{10}$$

Grouping together all the structures mentioned above, the deep learning based traffic flow prediction DNN-BTF model proposed in this study is trained in an end-to-end fashion. Specifically, Adamax optimizer (Kingma and Ba, 2014), which is a special stochastic gradient descend (SGD) method, is used to train the DNN-BTF model. The mean squared error in Eq. (11) is used as the objective function for DNN-BTF,

$$error = E[(\theta^p-\theta^t)^2], \tag{11}$$

where $\theta^p$ denotes the prediction value or outputs of the DNN-BTF model, $\theta^t$ is the true value or labeled data.

## 4. Results

### 4.1. Evaluation on prediction results

In this section, the traffic flow data from PeMS were used to evaluate the performance of the proposed DNN-BTF model through comparing the proposed model to several state-of-the-art forecasting methods with deep architectures. All of these experiments are performed using a NEW ALIENWARE 15 (CPU: i7-6700HQ, GPU: NVIDIA GeForce GTX 1060 6 GB).

#### 4.1.1. Experiments setup

The traffic flow data used in this study was from part of the North-bound Interstate 405, which is available at PeMS.[2] As is shown in Fig. 2, there were 33 detectors located along the I-405. between 04/01/2014 and 06/30/2015 given in are used in this study. As pointed out in Fusco and Gori (1995), a update time of 5 min is required to ensure traveler information systems be beneficial in

[2] http://pems.dot.ca.gov/.

congested transportation networks. Thus the most common used traffic flow prediction experiments procedure (Zhang et al., 2014; Qi and Ishak, 2014; Fusco et al., 2016), were adopted in this study, where the traffic volumes are aggregated into 5 min interval. Consequently, one detector preserves 288 data points per day. The proposed method was evaluated on a multi-horizon prediction task: the time window size $n$ of $\mathbf{S}^{\mathbf{f}}$ is set as 21. The prediction horizon $h$ is set as 9, which means that 105 min historical data are used to perform the traffic flow forecasting of the next 45 min. As an example, suppose that the current time is 6:30 PM, and that the flow at (6:35 PM, 6:40 PM, …, 7:15 PM) are to be predicted, the process continues until the flow at last 45 min of the datasets are predicted. The time lags of daily periodicity $n^d$ and weekly periodicity $n^w$ for long-term inputs of the DNN-BTF model are set as 6, which means that the traffic flow 30 min before and after the prediction horizon in previous day and in previous week are used as periodic inputs (the size of daily windows and weekly windows is also 21). The past-future traffic flow pairs from 0:00 AM 04/01/2014 to 6:30 PM 06/20/2015 are used to train prediction models, the rest data from 6:30 PM 06/20/2015 to 24:00 PM 06/30/2015 are used for evaluation. The number of training samples is 100,000 All prediction models are simultaneously trained to forecast the traffic flow at those 33 detector locations. The aggregated average traffic volume at 288 time points of a day, the aggregated average daily traffic of a week, and the average traffic volume of the locations at those 33 detectors are shown in Fig. 3. It is known that traffic flow data exhibit morning and evening peaks on a daily basis, and the volume of traffic flow of workdays is higher than that of weekends.

The proposed DNN-BTF model was compared with several state-of-the-art methods, which include the following:

- LASSO (Kamarianakis et al., 2012),
- traditional shallow back-propagation neural network (BPNN),
- stacked autoencoder (SAE) (Lv et al., 2015),
- DeepST (Zhang et al., 2016b), and
- Sequence to Sequence learning (StoS).

Those methods include prediction methods with different structures and depths. Slight modifications have been made to accommodate the test datasets to those methods and to ensure the fairness of these experiments. The settings of all these aforementioned methods are given as following.

To guarantee the fairness of these experiments on neural network approaches, they are all trained by Adamax optimizer (Kingma and Ba, 2014) with a batch size of 300. we use 10% training data as validation set to avoid overfitting. Three 1D convolutional layers with SReLU activation were used to extract the spatial features for the DNN-BTF model. The larger the number of feature maps is, the longer the training and prediction times are. The number of feature maps of each layer of the CNN is fixed as 30 to balance the computational cost and prediction accuracy. To determine the filtering lengths of the CNNs in the DNN-BTF model, values from 2 to 5 were tested in each layer given 30 feature maps. It is found that these CNNs with the filter lengths 4, 3, and 2 for the 1st, the 2nd, and the 3rd layers achieve the lowest error rates. Meanwhile, stacked GRUs with 2 layers were used to extract the temporal features. The dimensions of hidden states of all GRUs were set as 50. A single layer neural network with 600 hidden neurons using RelU activation was used as the attention model to determine the importance of the near-term inputs of traffic flow. 50 states and 600 hidden neurons were chosen in order to balance the computation cost and prediction accuracy. To give a fair comparison, parameters of these compared methods were chosen according to the results in literature and the experimental results of the proposed DNN-BTF model. Some chosen key parameters include: The BPNN method uses a single layer neural network with 1900 hidden neurons using RelU activation. For SAE, a three-depth deep neural network are used, we group all the inputs mentioned in above sections together and input them to the SAE and the BPNN, the hidden neuron for all the hidden layer is 300, and the activation is the sigmoid function, each layer is pre-trained by the autoencoder approach. In the DeepST method, the original 2D convolutional layers were modified as 1D convolutional layers. The SReLU activation is used as activation of convolutional layer. The weather inputs in original work were not used in this comparison. More details of the DeepST method can be found in Zhang et al. (2016b). The proposed architecture in Cho et al. (2014) were used in the StoS method, which is originally applied to language translation. Two layers of Gated Recurrent neural network (GRU) are used in the StoS method. The dimension of the hidden outputs for the first GRU is set as 50. In the LASSO model, the weight of $l_1$ norm loss is set as 0.02. All neural network approaches are built upon keras[3] framework and the LASSO model is implemented on scikit-learn[4] framework. The traffic flow and speed are normalized to be between 0 and 1 to train all the model.

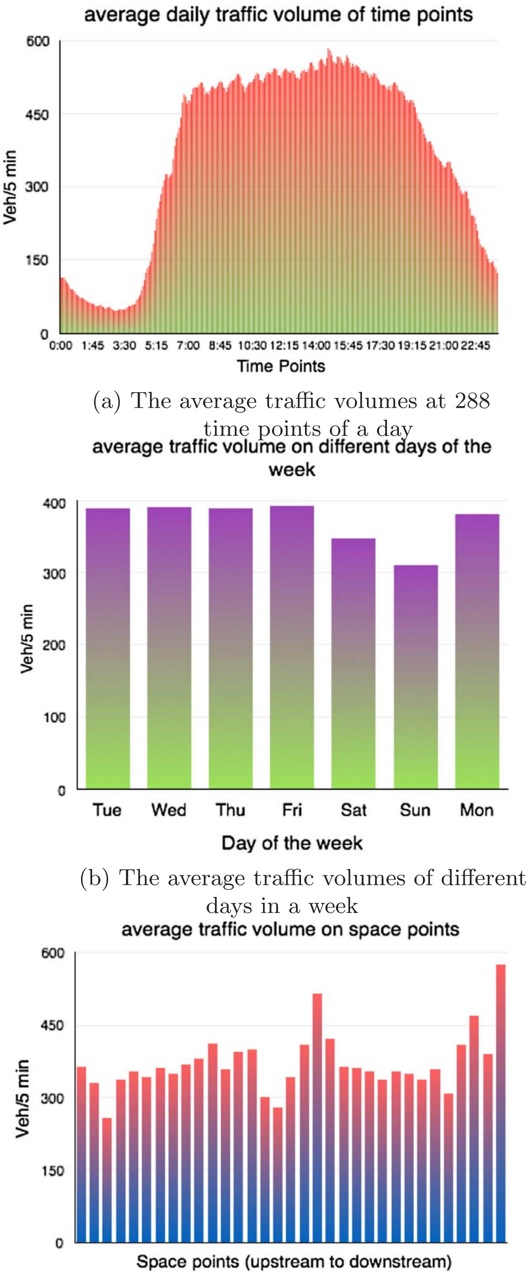
### 4.1.2. Prediction accuracy comparisons

In general, the mean relative error (MRE) index is one of the most common measurements to compare accuracy of different prediction methods. However, the MRE will be lower if the traffic volumes are higher. In observance of this, this paper also applies the mean absolute error (MAE) and root mean square error (RMSE) as a complementary measure for MRE. The MAE, MRE and RMSE are defined as following:

$$MAE = \frac{\sum_i^I |P_i - T_i|}{I}, \tag{12}$$

[3] https://github.com/fchollet/keras.

[4] http://scikit-learn.org/stable/.



(a) The average traffic volumes at 288 time points of a day

(b) The average traffic volumes of different days in a week

(c) The average traffic volumes at the locations of those 33 detectors

**Fig. 3.** The average traffic volumes: (a) at 288 time points of a day, (b) of different days in a week, (c) and at the locations of those 33 detectors.

$$MRE = \frac{\sum_i^I |P_i - T_i|/T_i}{I}, \tag{13}$$

$$RMSE = \sqrt{\frac{\sum_i^I |P_i - T_i|^2}{I}}, \tag{14}$$

$I$ denotes the total number of prediction point, $P_i$ is the prediction value, $T_i$ is the true value.

Table 1 presents the error indexes of different methods at different future time points. It is found that the improvement of DNN-BTF model can be observed at all time points. The proposed DNN-BTF model works best at one time point ahead of the prediction, which outperforms the second performer BPNN with improvements of 1.9411 and 2.2048 on MAE and RMSE. The DNN-BTF achieves improved average performance on MAE, RMSE and MRE compared to other methods, which outperforms the BPNN with

**Table 1**
The MAEs, MREs, and RMSEs on different prediction horizons ($h = 9$).

| Time points | Error indexes | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| DNN-BTF | MAE | **19.1264** | **20.8980** | **22.1413** | **22.5147** | 23.2682 | **23.8734** | **24.2935** | **24.8849** | **25.1895** |
| | MRE | **0.0700** | **0.0784** | **0.0787** | **0.0806** | **0.0819** | **0.0837** | **0.0856** | **0.0873** | **0.0896** |
| | RMSE | **27.9183** | **30.3107** | **32.1639** | **32.8138** | **33.7022** | **34.2854** | **34.8114** | **35.6132** | **36.0844** |
| LASSO | MAE | 22.3125 | 24.1387 | 25.3378 | 26.3254 | 27.0936 | 27.8473 | 28.5432 | 29.1087 | 29.5867 |
| | MRE | 0.0999 | 0.1074 | 0.1126 | 0.1171 | 0.1207 | 0.1245 | 0.1284 | 0.1317 | 0.1350 |
| | RMSE | 31.1657 | 34.2192 | 36.2820 | 37.9371 | 39.2922 | 40.6149 | 41.9039 | 42.9935 | 43.8450 |
| BPNN | MAE | 21.0675 | 22.3222 | 23.1900 | 23.6877 | 23.9638 | 24.28610 | 24.5079 | 24.9513 | 25.6721 |
| | MRE | 0.0782 | 0.0821 | 0.0850 | 0.0869 | 0.0892 | 0.0914 | 0.0930 | 0.0952 | 0.0983 |
| | RMSE | 30.1231 | 32.1101 | 33.3889 | 34.0898 | 34.5754 | 35.1769 | 35.6520 | 36.3148 | 37.2048 |
| SAE | MAE | 21.9578 | 23.0995 | 23.8966 | 24.6171 | 25.3099 | 25.6771 | 25.8733 | 26.7037 | 27.1733 |
| | MRE | 0.0890 | 0.0942 | 0.0992 | 0.0994 | 0.1001 | 0.1024 | 0.1045 | 0.1079 | 0.1082 |
| | RMSE | 31.4613 | 33.1703 | 34.2818 | 35.4368 | 36.4692 | 37.0338 | 37.2751 | 38.2572 | 38.8288 |
| DeepST | MAE | 21.4280 | 22.6632 | 23.3662 | 24.0226 | 24.6017 | 25.2731 | 25.8846 | 26.4139 | 27.1316 |
| | MRE | 0.0742 | 0.0773 | 0.0828 | 0.0849 | 0.0879 | 0.0932 | 0.0933 | 0.0954 | 0.0991 |
| | RMSE | 29.8473 | 31.7386 | 32.8924 | 33.8946 | 34.7322 | 35.6000 | 36.4113 | 37.1146 | 38.0519 |
| StoS | MAE | 22.2195 | 23.1367 | 23.8376 | 24.4685 | 25.0457 | 25.6200 | 26.1876 | 26.7975 | 27.5627 |
| | MRE | 0.0898 | 0.0942 | 0.0973 | 0.1013 | 0.1055 | 0.1093 | 0.1127 | 0.1158 | 0.1194 |
| | RMSE | 31.3893 | 32.6846 | 33.6911 | 34.4572 | 35.1251 | 35.7961 | 36.4985 | 37.2847 | 38.2864 |

improvements of 0.8287 and 1.2148 on MAE and RMSE, and outperforms the DeepST with improvement of 0.0582 on MRE. The experimental results confirms that our approach can learn rich yet compact spatial-temporal feature within traffic flow. There is a clear phenomenon that each method performs better in near-term future time point as prediction in a near future is easier. All neural networks based methods perform better than LASSO, the reason is that LASSO is a simple linear model which cannot sufficiently capture the nonlinear features within traffic flow. A surprising finding in our experiments is that single layer BPNN even perform better than other deep learning methods except our DNN-BTF. There may be two reasons:

- Compared with SAE methods, more advanced activation function ReLU is used for BPNN, which makes it easier to optimize, and the deep fully connected architecture is very difficult to train even using the pre-training strategy.
- The regression layer is not employed in DeepST and StoS structures, it limits their efficient usage of spatial-temporal features of traffic flow.

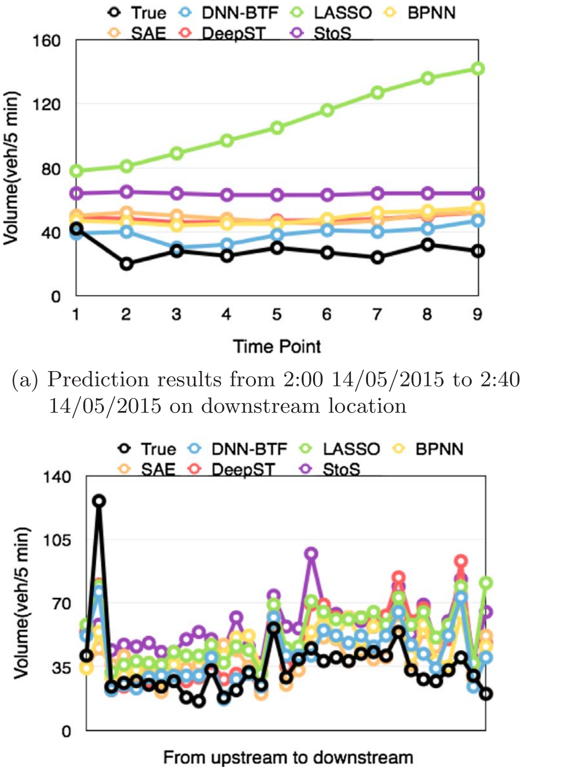
The aim of MAEs, MREs, and RMSEs is to measure the errors between the predicted values and the actual values. The forecasting accuracy of spatial and temporal distributions are also important indexes as we perform predictions at multiple locations and at time points. Thus, an average correlation (AC) is defined to measure the performance of spatial and temporal distribution forecasting:

$$
\begin{aligned}
AC(t) &= \frac{1}{n_t}\sum_{t=1}^{n} Corr(P_{:t},T_{:t}),\\
ACE(s) &= \frac{1}{n_s}\sum_{s=1}^{n} Corr(P_{s:},T_{s:}),
\end{aligned}
\tag{15}
$$

where $P_{:t}$ and $P_{s:}$ predict traffic flow vector at time point $t$, and space point $s$, respectively; $T_{:t}$ and $T_{s:}$ denote actual traffic flow vectors; $n_t$ ,and $n_s$ are the number of predicted vectors on time dimension, and space dimension respectively. Table 2 gives the average correlation of all of these methods on time and space dimensions. The AC comparisons reveal that the proposed DNN-BTF method can provide reliable prediction in terms of space and time distributions of traffic flow. Interestingly, DeepST- a full convolutional network

**Table 2**
The ACs of different methods.

| Method | AC(s) | AC(t) |
|---|---|---|
| DNN-BTF | 0.8293 | 0.4043 |
| LASSO | 0.8160 | 0.3697 |
| BPNN | 0.8260 | 0.4036 |
| SAE | 0.8006 | 0.4015 |
| DeepST | 0.8310 | 0.4020 |
| StoS | 0.8164 | 0.3130 |



(a) Prediction results from 2:00 14/05/2015 to 2:40 14/05/2015 on downstream location

(b) Prediction results on downstream location at 2:05 14/05/2015 (The second prediction point)

**Fig. 4.** Graphical illustration of prediction results.

exhibits the highest spatial average correlation compared with other methods including DNN-BTF. It indicates that convolutional network has a powerful capability to extract the space features within traffic flow. However, it will be more useful if conjoined with other architectures as the proposed DNN-BTF model. Fig. 4 gives a graphical illustration of prediction results produced by these methods, it is clear DNN-BTF is more accurate in both temporal and spatial dimension on this prediction period.

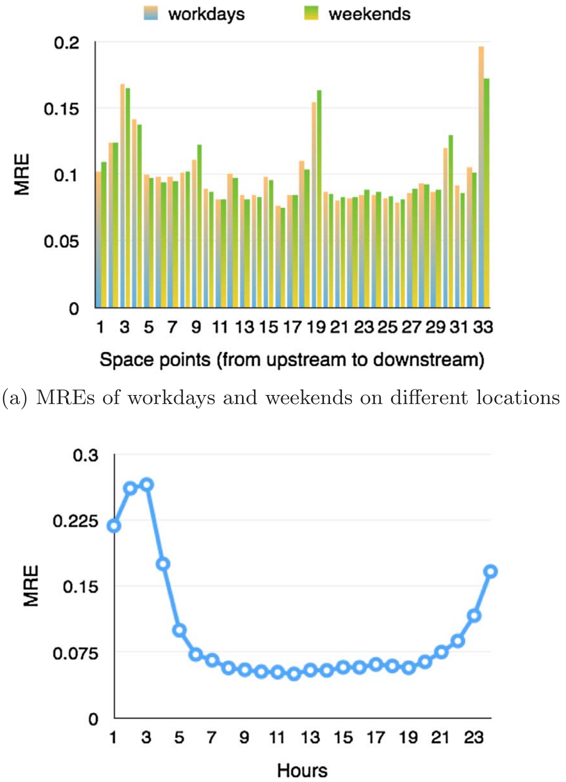
## 4.2. *Visualization and understanding*

In this section we conduct a direct analysis of the information hidden in the “brain” of the DNN-BTF by asking the following question: Give an well-trained deep neural network, to which extent is it to understand the phenomenon hidden in traffic data?

### 4.2.1. *Error analysis*

The prediction performance of all of these models vary with many conditions including time-of-day, day-of-week, and spatial locations. To understand how various conditions influence the prediction performance, a statistical analysis is conducted to examine the prediction capability of the DNN-BTF model. The results are presented in Fig. 5. Three different conditions are considered: weekdays/weekends, prediction time, and spatial points. The prediction time is set according to $t$ in Eq. (1). For example, the prediction time is set as 7 if $t$ belongs to 7:00 AM. The results are aggregated over all 45 min prediction horizon.

As shown in Fig. 5(a), the MREs on workdays and weekends are not significantly different. This result indicates that the performance of DNN-BTF is stable with respect to day-of-week. In addition, the proposed DNN-BTF method is shown to exhibit high MREs at the 3rd, the 19th, and the last detector locations. It is also observed from Fig. 3 that the average traffic volume at the 3rd location is the lowest, and the 19th and last locations exhibit the highest traffic volume among all those 33 locations. This result may be explained by the fact that the traffic flow patterns of those locations are significantly different with other patterns. Same results were observed from other tested models. Moreover, it is evident that DNN-BTF method exhibits stronger prediction capability on daytime (see Fig. 5(b)). There is little difference on MREs of rush-hour and of non-rush hours. The average MREs are about 0.06 during daytime. However, the average MREs are above 0.2 between 1:00 and 3:00 AM. As shown in Fig. 1, the traffic volume is very low during 1:00 and 3:00 AM, which means that traffic flow are more likely to exhibit free flow state during that time period. Nevertheless, the performance of the DNN-BTF method during 1:00 and 3:00 AM are consistent with the performances of other compared models.



(a) MREs of workdays and weekends on different locations

(b) MREs on different prediction hours

**Fig. 5.** Error analysis on DNN-BTF.

#### *4.2.2. Attention model: learning to attend*

By analyzing the attention scores learned by the attention model described in Section 3.1, we are able to learn the view of the DNN-BTF method on propagation mechanism of traffic flow. To further understand the propagation mechanism learned by the attention model used in the DNN-BTF method, the evolution of attention scores were analyzed with respect to time lags, space point, traffic speed, and traffic flow.

The results are shown in Fig. 6. In time dimension, the average score gradually increases when the time lags are above $-7$. The attention model gives a very high average score 0.879 when time lag is $-1$. In this respect, the attention learned by DNN-BTF is very similar to the prediction approaches using rules designed by humans to determine the weights. It generally depends heavier on traffic flow between recent 35 min (7 points) time period data to forecast future traffic flow. However, the attention outputs in longer time lags are very complex, it does not monotonically increase with time lags. In space dimension, it is harder to find significant regularity, but DNN-BTF attend more to upstream points and downstream points than to intermediate points. However, why it gives a higher average accuracy on some space points to calculate future traffic flow remain needs further studies, it gives highest attention to 8th location which is an approximate intermediate point between on-ramps and off-ramps, while points with the second and third high scores are near off-ramps. We further calculate the average values of attention score with respect to different traffic speed-flow horizons. It is found that attention model of DNN-BTF gives the highest attention score when traffic speed is very fast and traffic flow is low, which is corresponding to free flow state. In this state, traffic flow is smooth and the time that traffic flow exhibit such phenomenon is always corresponding to non-peak hours, DNN-BTF learn that traffic flow between long time lags and in points with long distance are more likely to impact future traffic flow in this state as vehicles can drive fast and tend to drive for a longer distance.

#### *4.2.3. Visualizing deep space-representation*

In this subsection, we conduct a direct analysis of spatial features captured by CNNs through characterizing the information that they retain layer by layer. As common works on visual task, we do so by modeling features in each layer of CNN as a function $f(x)$ of the input traffic flow and then reconstruct $x$ from it. In our analysis, the traffic flow is reconstructed by optimizing the objective

$$x_n^* = argmin_{x_n^*} \|f_n(x) – f_n(x_n^*)\|^2, \tag{16}$$

in which $f_n(x)$ is the representation of $x$ in layer $n$ of a CNN, $x$ is the inputs, $x_n^*$ is the reconstructed representation from layer $n$. In proposed DNN-BTF, 3 3-layer CNNs are used to extract spatial features of near-term (CNN-n), daily periodic (CNN-d) and weekly periodic inputs (CNN-w), and the size of the inputs are same. Thus we can reconstruct same inputs using these CNNs and compare similarities and differences of their mechanism. The inputs of CNN-n network is slightly different from CNN-d and CNN-w, its inputs

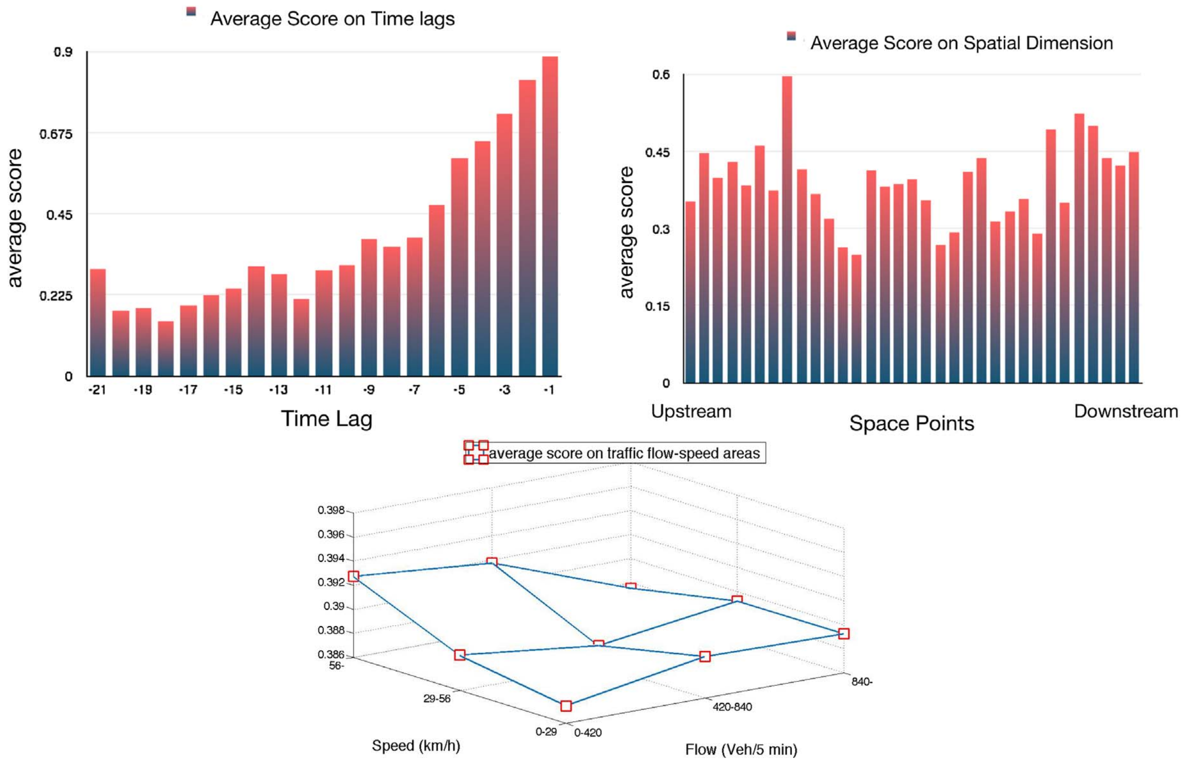


**Fig. 6.** The average scores at different time lags, at different space points, and at different traffic flow-speed areas, the scores are calculated by the average values that meet the set conditions.

is point-wise product with the outputs of attention network. We first reconstruct $x$, then point-wise divide $x$ with the corresponding attention score to obtain the reconstructed traffic flow. The reconstruction is achieved by using lbfgs algorithm built in scipy[5] to solve the optimization problem in Eq. (16).

Qualitatively, Fig. 7 illustrates the reconstruction of inputs between 0:00 AM and 1:45 AM on 17/05/2014 from each layer of CNN-n, CNN-d, and CNN-w. The progression with the depths of the layers is remarkable. The first layer maintains more information of the inputs. The deeper reconstruction contains less information and more meaningless points (above 1 and less than 0). CNN-n exhibits significant difference with CNN-d and CNN-w in terms of reconstruction accuracy and meaningless points distribution. Its reconstruction accuracy is worse than those of CNN-n and CNN-w, and its meaningful points are concentrated on later time points which is more close to prediction horizon. As for this phenomenon, there may be two reasons:

- The CNN-n works together with an attention model, the attention model forces the CNN-n to attend more to closer points as mentioned above;
- The traffic flow closer to prediction horizon contributes more to future traffic flow, while previous day's and the previous week's travel demands around the time point of the prediction horizon may all contribute to future traffic flow.

Clearly that the deeper representation of CNN captures progressively more sparse representation of space-time traffic flow, showing that the network captures just parts of the inputs, which evidently suffices for prediction. The CNN captures relevant traffic flow automatically, it is more sensible than those methods that use rule-based or that use statistical methods to determine whether the past space-time points are relevant with future traffic flow (Min and Wynter, 2011; Habtemichael and Cetin, 2016).

### 4.2.4. Visualizing deep time-representation

With regard to the temporal features captured by deep GRU, we are particularly interested in looking at the distributions of gate activations $R_q$ and $Z_q$ in the networks because they control how information passes through time. We compute the fraction of time that when $R_q$ is above 0.9 and $Z_q$ is less than 0.1. As mentioned in Section 3.3, the network tends to learn long-term memory if most values of $Z$ are small and most values of $R$ are large. We give the numbers of long-memory cells (the fraction $f(R_q > 0.9) > 0.5$ and $f(Z_q < 0.1) > 0.5$) for all GRUs in Table 3. It is found that deeper GRUs generally contain more long-term cells. In addition, there is no long-term cell in the first layers of GRU-n, GRU-d, and GRU-w. This result on traffic flow data is consistent with the result on language (Karpathy et al., 2015), in which the first layer barely use previous hidden state. Therefore a deeper gated RNN is needed to capture

[5] http://www.scipy.org/.

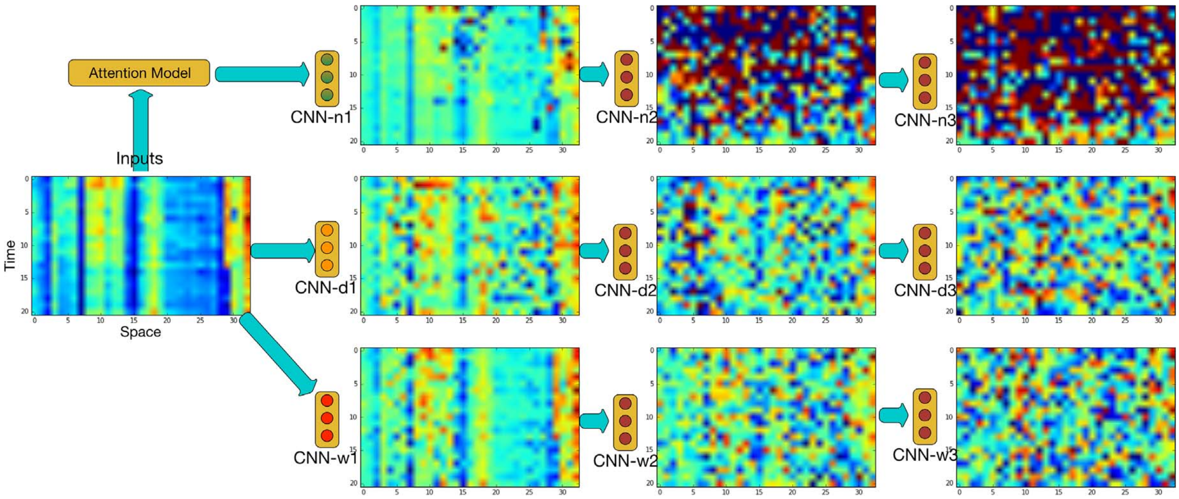


**Fig. 7.** CNN reconstruction. Reconstruction of the traffic flow from each layer of CNN-n, CNN-d, and CNN-w, the deep red denotes the value is above 1, the deep blue denotes the value is less than 0. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

**Table 3**
The numbers of long-memory cell: GRU-n: GRU with near-term inputs, GRU-d: GRU with daily periodic inputs, and GRU-w: GRU with weekly periodic inputs.

| Long-memory cell | $R_q$ | $Z_q$ |
|---|---|---|
| GRU-n Layer1 | 0 | 0 |
| GRU-n Layer2 | 1 | 1 |
| GRU-d Layer1 | 0 | 0 |
| GRU-d Layer2 | 4 | 2 |
| GRU-w Layer1 | 0 | 0 |
| GRU-w Layer2 | 3 | 1 |

the long-term memories within traffic flow. An interesting finding is that only a small part of deeper GRU cells are long-term memory cells, noted that the total numbers of $R_q$ and $Z_q$ are 50, the reason may be that a small part of cells is sufficient to capture the long-term memories within traffic flow data.

## 5. Conclusion and future work

This study contributes to the research on traffic flow predictions in three important ways:

- It proposes a new prediction model DNN-BTF, the model exploits the advantages of various deep learning architectures including fully-connected neural networks, recurrent neural networks, and convolutional neural networks to improve prediction performance;
- It introduces the attention model to traffic flow modeling, the attention model can automatically determine the importance of inputs of large scale space–time points;
- It conducts both qualitative and quantitative visualizations of the proposed DNN-BTF model. The attention score statistics, the space-feature reconstruction, and the memory cell statistics demonstrate that these networks learn powerful, and often interpretable past-future interactions on traffic flow data.

The developed model not only produced encouraging prediction accuracy, its visualization challenges the conventional thinking about neural networks in the transportation field that neural networks is purely a “black-box” model. To the best of our knowledge, most existing traffic flow prediction researches focus on searching for models with higher prediction accuracy, this work not only proposes a prediction model with reliable accuracy, but also analyzes the internal mechanism of DNN on traffic flow data. It suggests a new way of thinking about both traffic flow prediction and traffic flow data analytics.

However, as the results demonstrate, the deep learning based traffic flow prediction still need more future studies. In particular, more efficient deep architectures combined with traffic flow theory and their application on urban transportation networks are very worth studying. The understanding and visualization of deep learning on traffic flow will be another future research direction. In fact, deep learning is very functional for other transportation related applications such as missing data imputation and traffic event detection.

## Acknowledgement

The work was supported by NSFC (61620106002, 61271376). The authors would like to thank experienced anonymous reviewers for their constructive and valuable suggestions for improving the overall quality of this paper. The authors acknowledge the help of Jacci Ziebert, who has offered suggestions for improving writing quality of this paper.

## References

Allström, A., Ekström, J., Gundlegård, D., Ringdahl, R., Rydergren, C., Bayen, A.M., Patire, A.D., 2016. Hybrid approach for short-term traffic state and travel time prediction on highways. Transport. Res. Rec.: J. Transport. Res. Board (2554), 60–68.

Bengio, Y., 2009. Learning deep architectures for ai. Found. Trends® Mach. Learn. 2 (1), 1–127.

Cho, K., Van Merriënboer, B., Bahdanau, D., Bengio, Y., 2014. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. Available from: < arXiv:1409.1259 > .

Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T., 2015. Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2625–2634.

Duan, Y., Lv, Y., Wang, F.-Y., 2016. Travel time prediction with lstm neural network. In: Intelligent Transportation Systems (ITSC), 2016 IEEE 19th International Conference on. IEEE, pp. 1053–1058.

Ermagun, A., Chatterjee, S., Levinson, D., 2017. Using temporal detrending to observe the spatial correlation of traffic. PLoS One 12 (5), e0176853.

Fusco, G., Colombaroni, C., Isaenko, N., 2016. Short-term speed predictions exploiting big data on large urban road networks. Transport. Res. Part C: Emerg. Technol. 73, 183–201.

Fusco, G., Gori, S., 1995. The use of artificial neural networks in advanced traveler information and traffic management systems. Appl. Adv. Technol. Transport. Eng.: ASCE 341–345.

Guo, J., Huang, W., Williams, B.M., 2014. Adaptive kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification. Transport. Res. Part C: Emerg. Technol. 43, 50–64.

Habtemichael, F.G., Cetin, M., 2016. Short-term traffic flow rate forecasting based on identifying similar traffic patterns. Transport. Res. Part C: Emerg. Technol. 66, 61–78.

Henaff, M., Bruna, J., LeCun, Y., 2015. Deep convolutional networks on graph-structured data. Available from: < arXiv:1506.05163 > .

Hinton, G., Deng, L., Yu, D., Dahl, G.E., Mohamed, A.-R., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T.N., et al., 2012. Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. IEEE Signal Process. Magaz. 29 (6), 82–97.

Huang, W., Song, G., Hong, H., Xie, K., 2014. Deep architecture for traffic flow prediction: deep belief networks with multitask learning. IEEE Trans. Intell. Transport. Syst. 15 (5), 2191–2201.

Jiang, X., Adeli, H., 2005. Dynamic wavelet neural network model for traffic flow forecasting. J. Transport. Eng. 131 (10), 771–779.

Jin, P.J., Yang, F., Cebelak, M., Ran, B., Walton, C., 2013. Urban travel demand analysis for austin tx usa using location-based social networking data. In: TRB 92nd Annual Meeting Compendium of Papers.

Jin, X., Xu, C., Feng, J., Wei, Y., Xiong, J., Yan, S., 2015. Deep learning with s-shaped rectified linear activation units. Available from: < arXiv:1512.07030 > .

Kamarianakis, Y., Shen, W., Wynter, L., 2012. Real-time road traffic forecasting using regime-switching space-time models and adaptive lasso. Appl. Stoch. Models Bus. Indus. 28 (4), 297–315.

Karlaftis, M.G., Vlahogianni, E.I., 2011. Statistical methods versus neural networks in transportation research: differences, similarities and some insights. Transport. Res. Part C: Emerg. Technol. 19 (3), 387–399.

Karpathy, A., Johnson, J., FeiFei, L., 2015. Visualizing and understanding recurrent networks. Available from: < arXiv:1506.02078 > .

Kingma, D., Ba, J., 2014. Adam: A Method for Stochastic Optimization. Available from: < arXiv:1412.6980 > .

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. Adv. Neural Inform. Process. Syst. 1097–1105.

Kumar, S.V., Vanajakshi, L., 2015. Short-term traffic flow prediction using seasonal arima model with limited input data. Eur. Transp. Res. Rev. 7 (3), 1–9.

Li, L., Li, Y., Li, Z., 2013. Efficient missing data imputing for traffic flow by considering temporal and spatial dependence. Transport. Res. Part C: Emerg. Technol. 34, 108–120.

Lippi, M., Bertini, M., Frasconi, P., 2013. Short-term traffic flow forecasting: an experimental comparison of time-series analysis and supervised learning. IEEE Trans. Intell. Transport. Syst. 14 (2), 871–882.

Lopez-Garcia, P., Onieva, E., Osaba, E., Masegosa, A.D., Perallos, A., 2016. A hybrid method for short-term traffic congestion forecasting using genetic algorithms and cross entropy. IEEE Trans. Intell. Transport. Syst. 17 (2), 557–569.

Lv, Y., Duan, Y., Kang, W., Li, Z., Wang, F.-Y., 2015. Traffic flow prediction with big data: a deep learning approach. IEEE Trans. Intell. Transport. Syst. 16 (2), 865–873.

Ma, X., Tao, Z., Wang, Y., Yu, H., Wang, Y., 2015. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. Transport. Res. Part C: Emerg. Technol. 54, 187–197.

Mahendran, A., Vedaldi, A., 2015. Understanding deep image representations by inverting them. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, pp. 5188–5196.

Min, W., Wynter, L., 2011. Real-time road traffic prediction with spatio-temporal correlations. Transport. Res. Part C: Emerg. Technol. 19 (4), 606–616.

Nair, V., Hinton, G.E., 2010. Rectified linear units improve restricted Boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10), pp. 807–814.

Okutani, I., Stephanedes, Y.J., 1984. Dynamic prediction of traffic volume through kalman filtering theory. Transport. Res. Part B: Methodol. 18 (1), 1–11.

Peris, Á., Bolaños, M., Radeva, P., Casacuberta, F., 2016. Video description using bidirectional recurrent neural networks. Available from: < arXiv:1604.03390 > .

Polson, N., Sokolov, V., 2016. Deep learning predictors for traffic flows. Available from: < arXiv:1604.04527 > .

Qi, Y., Ishak, S., 2014. A hidden markov model for short term prediction of traffic conditions on freeways. Transport. Res. Part C: Emerg. Technol. 43, 95–111.

Ran, B., Jin, P.J., Boyce, D., Qiu, T.Z., Cheng, Y., 2012. Perspectives on future transportation research: impact of intelligent transportation system technologies on next-generation transportation modeling. J. Intell. Transport. Syst. 16 (4), 226–242.

Shahsavari, B., Abbeel, P., 2015. Short-term traffic forecasting: modeling and learning spatio-temporal relations in transportation networks using graph neural networks.

Simonyan, K., Vedaldi, A., Zisserman, A., 2013. Deep inside convolutional networks: Visualising image classification models and saliency maps. Available from: < arXiv:1312.6034 > .

Socher, R., Huval, B., Bath, B., Manning, C.D., Ng, A.Y., 2012. Convolutional-recursive deep learning for 3d object classification. Adv. Neural Inform. Process. Syst. 665–673.

Springenberg, J.T., Dosovitskiy, A., Brox, T., Riedmiller, M. 2014. Striving for simplicity: The all convolutional net. Available from: < arXiv:1412.6806 > .

Sun, H., Liu, H., Xiao, H., He, R., Ran, B., 2003. Use of local linear regression model for short-term traffic forecasting. Transport. Res. Rec.: J. Transport. Res. Board (1836), 143–150.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R., 2013. Intriguing properties of neural networks. Available from: < arXiv:1312.6199 > .

Tan, H., Wu, Y., Shen, B., Jin, P.J., Ran, B., 2016a. Short-term traffic prediction based on dynamic tensor completion. IEEE Trans. Intell. Transport. Syst. 17 (8), 2123–2133. http://dx.doi.org/10.1109/TITS.2015.2513411.

Tan, H., Feng, G., Feng, J., Wang, W., Zhang, Y.-J., Li, F., 2013a. A tensor-based method for missing traffic data completion. Transport. Res. Part C: Emerg. Technol. 28, 15–27.

Tan, H., Feng, J., Feng, G., Wang, W., Zhang, Y.-J., 2013b. Traffic volume data outlier recovery via tensor model. Mathematical Problems in Engineering 2013.

Tan, H., Xuan, X., Wu, Y., Zhong, Z., Ran, B., 2016b. A comparison of traffic flow prediction methods based on dbn. CICTP 2016. 273–283.

Vinyals, O., Toshev, A., Bengio, S., Erhan, D., 2015. Show and tell: A neural image caption generator. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3156–3164.

Vlahogianni, E.I., Karlaftis, M.G., Golias, J.C., 2014. Short-term traffic forecasting: where we are and where were going. Transport. Res. Part C: Emerg. Technol. 43, 3–19.

Wang, J., Shi, Q., 2013. Short-term traffic speed forecasting hybrid model based on chaos-wavelet analysis-support vector machine theory. Transport. Res. Part C: Emerg. Technol. 27, 219–232.

Wang, J., Yu, L.-C., Lai, K.R., Zhang, X., 2016. Dimensional sentiment analysis using a regional cnn-lstm model. In: The 54th Annual Meeting of the Association for Computational Linguistics, p. 225.

Wu, Y., Tan, H., Li, Y., Li, F., He, H., 2017. Robust tensor decomposition based on cauchy distribution and its applications. Neurocomputing 223, 107–117.

Wu, Y., Tan, H., Peter, J., Shen, B., Ran, B., 2015a. Short-term traffic flow prediction based on multilinear analysis and k-nearest neighbor regression. In: 15th COTA International Conference of Transportation Professionals.

Wu, Z., Wang, X., Jiang, Y.-G., Ye, H., Xue, X., 2015b. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In: Proceedings of the 23rd ACM international conference on Multimedia. ACM, pp. 461–470.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., Zemel, R.S., Bengio, Y., 2015. Show, attend and tell: Neural image caption generation with visual attention. Available from: < arXiv:1502.03044 > 2(3) 5.

Xu, X.-y., Liu, J., Li, H.-y., Jiang, M., 2016. Capacity-oriented passenger flow control under uncertain demand: algorithm development and real-world case study. Transport. Res. Part E: Logist. Transport. Rev. 87, 130–148.

Yang, H.-F., Dillon, T.S., Chen, Y.-P.P., 2016. Optimized structure of the traffic flow forecasting model with a deep learning approach. IEEE Trans. Neural Networks Learn. Syst.

Yang, S., Shi, S., Hu, X., Wang, M., 2015. Spatiotemporal context awareness for urban traffic modeling and prediction: sparse representation based variable selection. PloS One 10 (10), e0141223.

Yao, L., Torabi, A., Cho, K., Ballas, N., Pal, C., Larochelle, H., Courville, A., 2015. Describing videos by exploiting temporal structure. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4507–4515.

Yosinski, J., Clune, J., Nguyen, A., Fuchs, T., Lipson, H., 2015. Understanding neural networks through deep visualization. Available from: < arXiv:1506.06579 > .

Yuan, J., Zheng, Y., Xie, X., Sun, G., 2011. Driving with knowledge from the physical world. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, pp. 316–324.

Zeiler, M.D., Fergus, R., 2014. Visualizing and understanding convolutional networks. In: European Conference on Computer Vision. Springer, pp. 818–833.

Zhang, J., Zheng, Y., Qi, D., 2016a. Deep spatio-temporal residual networks for citywide crowd flows prediction. Available from: < arXiv:1610.00081 > .

Zhang, J., Zheng, Y., Qi, D., Li, R., Yi, X., 2016b. Dnn-based prediction model for spatio-temporal data.

Zhang, Y., Haghani, A., 2015. A gradient boosting method to improve travel time prediction. Transport. Res. Part C: Emerg. Technol. 58, 308–324.

Zhang, Y., Zhang, Y., Haghani, A., 2014. A hybrid short-term traffic flow forecasting method based on spectral analysis and statistical volatility model. Transport. Res. Part C: Emerg. Technol. 43, 65–78.

Zheng, W., Lee, D.-H., Shi, Q., 2006. Short-term freeway traffic flow prediction: Bayesian combined neural network approach. J. Transport. Eng. 132 (2), 114–121.

Zheng, X., Chen, W., Wang, P., Shen, D., Chen, S., Wang, X., Zhang, Q., Yang, L., 2016. Big data for social transportation. IEEE Trans. Intell. Transport. Syst. 17 (3), 620–630.

Zintgraf, L.M., Cohen, T.S., Welling, M., 2016. A new method to visualize deep neural networks. Available from: < arXiv:1603.02518 > .