Multimedia Tools and Applications (2024) 83:7379–7394
https://doi.org/10.1007/s11042-023-15395-w

Attention based convolutional networks for traffic flow prediction

Juncong Lin[1] · **Chengqiao Lin**[1] · **Qi Ye**[2]

Received: 27 December 2021 / Revised: 22 April 2022 / Accepted: 18 April 2023 /
Published online: 8 June 2023

Abstract

Real-time and accurate prediction of traffic flow plays an important role in intelligent transportation systems. However, short-term traffic flow forecasting is extremely challenging due to the highly nonlinear nature of the traffic system and the dynamic spatial and temporal correlation. Although various methods, including deep learning based ones, have been proposed, most of them still suffer from problems such as spatial nonstationarity and thus cannot achieve good prediction performance. Inspired by the recent superior performance of attention mechanism, we introduce it into the model for traffic flow prediction with regular grided input. To be specific, we propose a novel deep learning framework, Spatial-Temporal Attention Based Convolutional Networks (STAtt-Net), for accurate forecasting of citywide traffic flow. First, we model the traffic data as a two-dimensional matrix with two channels. Each cell in the matrix represents the traffic in the corresponding region. Taking into account the temporal correlation and dependence of traffic system, the periodic patterns contained in traffic data are modeled by three major components for weekly trend, daily periodicity, and hourly closeness respectively. Then, STAtt-Net employs a STBlock as the basis unit to learn temporal dependence and spatial dependence of traffic flow, taking advantage of attention mechanism. We conduct extensive experiments to evaluate the performance of our model on three real-world datasets (TaxiBJ, BikeNYC, TaxiSZ), with the results revealing better prediction accuracy and efficiency of the proposed model against existing ones.

Keywords Deep learning · Traffic flow prediction · Attention mechanism · Spatial nonstationarity

✉ Qi Ye
18957875526@163.com

Juncong Lin
jclin@xmu.edu.cn

Chengqiao Lin
linchengqiao@stu.xmu.edu.cn

[1] School of Informatics, Xiamen University, Xiamen, Fujian, China

[2] Information Center of Ningbo Human Resources and Social Security Bureau, Ningbo, Zhejiang, China

1 Introduction

Traffic forecasting is important for location-based applications such as intelligent transportation systems and urban planning [11]. Real-time accurate traffic flow prediction can help improve traffic efficiency and reducing traffic congestion in traffic control. Especially, in road peak periods and traffic accident-prone areas, accurate short-term traffic flow prediction can not only provide a judgment basis for travelers to choose the optimal path, but also provide strong data support for managers to formulate effective control measures and thus reduce traffic congestion.

In general, traffic states or events of a spatial unit (e.g., region and street) are not isolated, but influenced by its neighbors. This is a typical phenomenon of spatial dependency that has been extensively considered in current traffic prediction studies [4, 21, 22, 51]. The spatial dependency can be expressed by the first law of geography: "Everything is related to everything else, but near things are more related than distant things" [42]. For instance, traffic congestion states may propagate from one road to another due to rush hour, unexpected accidents, or unreasonable traffic management etc. Therefore, it is vital to consider spatial dependency among road segments. Besides, historical traffic flow data is also interdependent in temporal. As illustrated by Fig. 1, we can observe apparent similarity between two adjacent time periods (weeks) and even different days in a period. However, dynamic changes in spatio-temporal characteristics are random, can occur at any time, and thus difficult to capture. Thus, accurate traffic forecasting is challenging.

Numerous traffic prediction algorithms have been developed within the last few decades. The auto regressive integrated moving average (ARIMA) [28, 45] methods took advantage of repeating occurrences in temporal historical data. However the data is required to be smooth and continuous. And the prediction accuracy is usually limited for the complex spatial-temporal attributes of urban traffic. Machine learning models, such as k-nearest neighbor [56], support vector regression [49], only need enough historical data to automatically establish the nonlinear feature mapping relationships between input and output. But

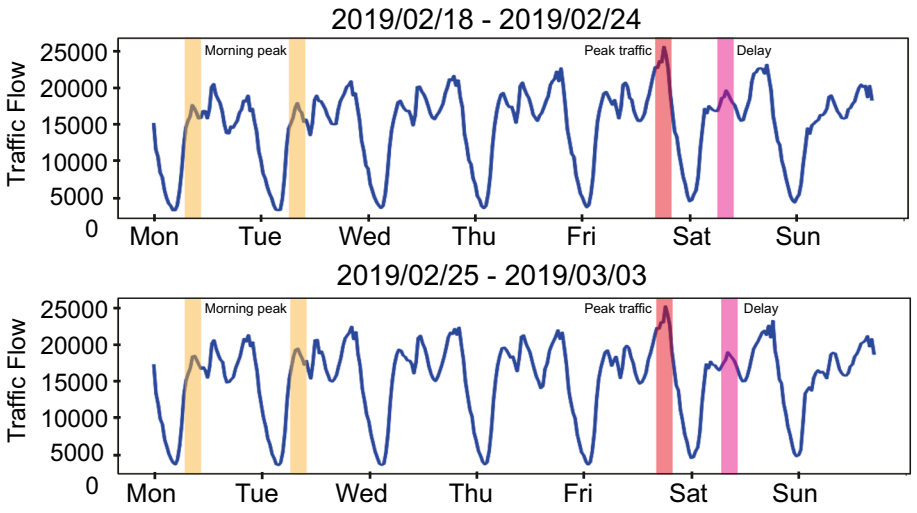

Fig. 1 Temporal distribution of Shenzhen taxi movements in two continuous weeks: from 02/18/2019(Monday) to 02/24/2019(Sunday) and from 02/25/2019(Monday) to 03/03/2019(Sunday)

the prediction accuracy of these methods is also not so satisfactory due to the difficulty in fully capturing of the complex nonlinear relationship. With the rise of deep neural networks (DNNs), many researchers also investigated the usage of DNNs in traffic prediction, either using CNNs [15, 57] to capture spatial dependencies or using LSTMs [8] to learn temporal depencence. However, the spatial topology of the traffic network may lose when the data is represented by matrix or multidimensional tensors. Therefore, even if CNN can extract spatial dependence, its effect on traffic flow prediction is still limited [23]. While LSTMs are computationally slow as they can not be trained in parallel, even though some acceleration methods [46] were proposed.

To tackle these challenges, we propose a new short-time traffic flow prediction framework based on deep learning. Attention mechanism is introduced in the framework to handle phenomenons that are hardly considered in previous methods such as spatial nonstationarity. To be specific, *STAtt-Net* incorporates a spatial-tempral attention model with the purpose to capture the global spatio-temporal by considering the interactions of region-to-region. In addition, we fuse three components: i) trend for weekly trend, ii) period for daily periodicity, and iii) closeness for recent time dependence together so as to capture temporal similarity more effectively.

The main contributions of our work are summarized as follows:

- A novel model, Spatio-Temporal Attention mechanism Network (*STAtt-Net*) for short-term traffic flow prediction, which can effectively exploits dynamic both temporal and spatial dependency in traffic.
- An attention based module *(STBlock)* for traffic prediction, capable of dynamic modeling the association between any two locations in a city.
- Extensive experiments with the proposed framework on three real-world datasets *TaxiBJ*, *BikeNYC* and *TaxiSZ*, with the experimental results revealing better performance of the proposed model over several state-of-the-art approaches.

2 Related work

2.1 Traffic flow prediction

Traffic flow prediction can be seen as a spatial-temporal forecast problem. Traditional methods targeting on this problem usually establish a time series model and exploit the relevant information hidden in the historical data for prediction. These methods can be categorized into parametric and nonparametric. Parametric approaches include autoregressive integrated moving average (ARIMA) model [7, 44], Kalman filtering (KF) [32], Structural time-series model(STM) [10] and latent space model [6] etc. However, these models rely on the stationary assumptions of traffic time series data and ignore the temporal and spatial dynamics. In order to deal with the stochastic and nonlinear nature of traffic data, researchers have paid much attention to the non-parametric approaches such as K-nearest neighbor(KNN) [3], Support Vector Regression(SVR) [36], Random Forest(RF) [2], etc. Unfortnuately, most non-parametric approaches are limited to model the complex dynamic spatial-temporal dependency. With the great success of deep learning in various applications such as computer vision [1, 17, 18, 29, 31, 37, 37, 40, 41], nature language processing [16], public health [30, 38, 47] and economy [19] etc, recent researches have leveraged deep learned features to further improve the performance of prediction by adopting various deep learning neural networks. An early attemp by Huang et al. [13] used a deep belief

networks (DBN) with multitask learning for traffic prediction. Although capble of mining high-dimensional features from traffic data, it is difficult to extract specific spatio-temporal features. Since RNNs are adept at extracting the correlation of temporal feature, it is unsurprisingly that many works in the area [8, 25] are built on RNN and it variants like Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU). For spatial dependency, CNN were introduced in ST-ResNet [57] to capture spatial correlation, combining with the residual unit for citywide traffic forecasting. Traffic flows were treated as a raster image to model the temporal closeness, period, trend, and external factors. And how to model nonlinear and complex spatial-temporal data simultaneously becomes a challenge. Shikhar et al. [34] employe 3D CNNs to recognize the patterns in volumetric data like videos, which proves the superior characteristics of 3D CNNs. Based on this characteristic, [9] apply 3D CNNs to automatically model saptio-temporal information and thus improve the accuracy of prediction. However, the improvement is limited due to the inefficient mining of spatio-temporal information. ConvLSTM [35] was proposed to settle spatio-temporal sequence forecasting problem, with a rather complex network structure. With the deepening of the network, training becomes more difficult. A major problem with current CNN based methods is that CNNs are suitable for Euclidean data (such as images, regular grids, etc.) but can not handle road networks with complex topology well. The recent work of Zeng et al. [53] revealed that when revisiting the modifiable area unit problem in deep traffic prediction, and tried to address the problem with deformable convolutions in the follow up [53].

This paper follows the work of Zeng et al. [53] to improve the performance of CNNs based model on complex road networks, but with a different strategy by introducing the attention mechanism.

2.2 Attention mechanism

Attention mechanisms is a recent popular topic, being widely used in different types of deep learning tasks such as natural language processing [24], image classification [27], machine translation [5] etc. Attention mechanism mimics the human brain's tendency to focus on something of interest and automatically ignore low-value information. Essentially, it is a combinatorial function that computes the probability distribution of attention to highlight the impact of a key input on the output, thereby achieving an efficient allocation of information processing resources. Mnih et al. [27] pioneered the use of attention mechanisms in image classification tasks and combined them with recurrent neural network models. Non-local Networks [43] utilizes self-attention as a non-local operation to capture long range dependencies. Yan et al. [48] proposed LVSNet for liver vessel segmentation, which employing an attention-guided concatenation module to enhances segmentation details. Hu et al. [12] proposed squeeze-and-excitation (SE) block to explicitly models the interdependencies between feature maps and adaptively obtains the importance of each feature map by learning. For traffic flow prediction, [50] designed a spatial-temporal dynamic network with a periodic transfer attention mechanism deal with capture long-term periodic temporal similarity. Zheng et al. [58] proposed a graph multi-attention network with spatial and temporal attentions. However the graph attention mechanism is only applicable to non-Euclidean structured data.

We design and integrate a spatio-temporal attention mechanism block to capture the dynamic relevance of the traffic network in the spatial and temporal dimensions respectively. Different from Zheng et al. [58], our mechanism works on regular grided map to handle Euclidean structured data.

3 Problem definition

We would like to firstly introduce some definitions and formulate the problem in this section before going into details of the method. The used notations are listed in Table 1.

Definition 1 (**Movement**) A *movement m* is a continuously measured trajectory of a moving object during a time period $\mathcal{T}$, which is defined by a set of spatio-temporal records $\bigcup_{t\in\mathcal{T}} < t, l_t >$, where l_t represents the position of m at time t. We denote all movements of multiple moving objects as $\mathcal{M}$.

Definition 2 (**Region**) There are many ways to partition a geographical area into a collection of appropriate regions $\mathcal{R} = \{r_k\}_{k=1}^{n}$. For example, a city can be partitioned into $n = I \times J$ equal-sized grids based on latitude and longitude, in which each grid is regarded as an independent spatial unit. Besides, according to census block information or function of different parts, the area can also be divided into non-overlaping and independent regions called traffic analysis zones (TAZ) [26].

Definition 3 (**Inflow & Outflow**) From the movements $\mathcal{M}$, we compute the inflow $x^{in}_{r_{i,j},t}$ and outflow $x^{out}_{r_{i,j},t}$ per time slot t for each region $r_{i,j} \in \mathcal{R}$ with the following formulas:

$$x^{in}_{r_{i,j},t} = \quad |\{m \in \mathcal{M} \mid m \cdot l_{t-1} \notin r_{i,j} \wedge m \cdot l_t \in r_{i,j}\}| \tag{1}$$

$$x^{\text{out}}_{r_{i,j},t} = \quad |\{m \in \mathcal{M} \mid m \cdot l_t \notin r_{i,j} \wedge m \cdot l_{t+1} \in r_{i,j}\}| \tag{2}$$

where $|\cdot|$ denotes the cardinality of the set. $x^{in}_{r_{i,j},t}$ and $x^{out}_{r_{i,j},t}$ indicates inflow and outflow at per time slot t for region $r_{i,j}$, respectively.

3.1 Problem formulation

Problem 1 (*Short-term traffic flow prediction*) Given a set of citywide historical traffic flow data represented by a series of matrices $\left\{X_{\mathcal{R},t} \mid t = 1, 2, \ldots, n\right\}$ in Region $\mathcal{R}$, the problem of traffic forecasting is to predict the traffic flow for all region cells in the next time interval $t + 1$, denote as $X_{\mathcal{R},t+1}$.

Table 1 Meanings of all notations

Notations	Description
$\mathcal{M}$; m	A collection of movements; a movement.
$\mathcal{T}$; t	A collection of time slots; a time slot.
$\mathcal{R}$; r	A collection of regions; a region.
$X_{\mathcal{R},t}$;	Aggregated traffic in regions $\mathcal{R}$ at time slot t;
$x_{r,t}$	Aggregated traffic in a region r at time slot t.
$\mathcal{G}$; g	Grid map; a grid.
$X_{\mathcal{G},t}$;	Aggregated traffic in grid map $\mathcal{G}$ at time slot t;
$x_{g,t}$	aggregated traffic in a grid g at time slot t.
$Y_{\mathcal{G},t+1}$;	Predicted result of grid map $\mathcal{G}$ at time slot $t + 1$;
$y_{g,t+1}$	Predicted result of a grid g at time slot $t + 1$.

4 Methodology

4.1 Data processing

Prior work [54] has demonstrated that the modifiable areal unit problem [33] within aggregation processes can lead to perturbations in the network inputs. As such, it eventually lead to inaccurate traffic flow forecasting results, affecting traffic planning decisions and other applications. We intend to further explore the effects of partition scale and manner on the prediction accuracy of of deep learning model in this work. We used the same three datasets (TaxiBJ, BikeNYC, and TaxiSZ) for experiments. Maps of the studying areas, (Beijing, New York and Shenzhen), need to be processed first for CNN input.

The map of Beijing is divided into 32×32 grids based on longitude and latitude. The data of the last four weeks in the dataset is kept as test set, and the rest is used for training. For *BikeNYC*, the entire city is break up into 8×16 grid map. The data of the last ten days in the data set is fetched for testing, and the rest data is used for training.

To explore the impact of different partition shapes, we use two types of partitioning to process the *TaxiSZ* dataset: TAZs and grids. TAZs are special zones usually designated by the department of transportation for tabulating traffic-related census data. A TAZ, a geographic grouping of census units, occupies a contiguous region with a minimum population of 600 in general. Besides, the border of a TAZ usually corresponds with recognizable physical boundaries, such as main streets and water sources. The land use activities and populations within each TAZ are relatively homogeneous. Thus TAZ partition can satisfy the need of traditional transportation planning and demand analysis better. However, the sizes of the regions are critical. Too small regions make the entry and departure between neighboring regions more frequent at various predictive times, introducing a lot of computational complexity. Larger regions, while greatly reducing the computational burden, are meaningless for traffic prediction purposes. We use the 491 TAZs provided by the Shenzhen Transportation Department. In order to better handle the traffic data in Shenzhen, we chose an appropriate size to divide Shenzhen into $\{r_k\}_{k=1}^{n}$ where $n = 1250$ for scale 25×50 and $n = 5000$ for scale 50×100 based on grid-based method. Same processing as before, we break each day into 48 slots with each slot lasting 30 min. As for TAZ-based method, we utilize official data from the city of Shenzhen to divide the city into $n = 491$ irregular regions. In this step, rasterization of TAZ partitions is a necessary step to fit in the inputs of our model.

Definition 4 (**Rasterization**) We divide TAZs into a grid map $\mathcal{G}$ of size $i \times j$. Each grid $g \in \mathcal{G}$ can intersect with arbitrary number of TAZs . We calculate the in/out traffic flow for each grid g at time slot t as:

$$x_{g,t} = \sum_{k=1}^{n} x_{r_k,t} \times \frac{S(r_k \cap g)}{S(r_k)}, \tag{3}$$

where $S(\cdot)$ stands for the area of a region, and $r_k \cap g$ indicates the intersection between r_k and g.

4.2 Spatial-temporal attention based convolutional networks

In order to model spatio-temporal dependency in traffic prediction, we design an end-to-end deep learning based model *STAtt-Net*. Figure 2 illustrates the overall framework, which consists of two major components: the temporal dependency module and the spatial attention block.

4.2.1 Temporal dependency module

It can be easily observed from the exemplary data in Fig. 1 that daily activities follows certain temporal periodicity in both day and week granularity, as pointed out by previous works [39, 50]. In particular, a morning peak can be found at around 9:00 every weekday, and however, the peaks delay about a half-hour every weekend. The traffic flow in a given arbitrary region is usually continuously varying, which means that the traffic flow at the current moment is strongly correlated with the traffic flow at the next moment. Therefore, the closeness for recent time dependency should be considered in traffic data prediction.

In addition, a travel peak is usually reached on Friday night. Based on this phenomenon, we consider the temporal dependency of hourly, daily and weekly. We set their length-dependent sequence Δh, Δd, Δw, among these three components, respectively. Thus our input data can be designed as:

$$X_h = \left[X_{\mathcal{G},t-\Delta h}, X_{\mathcal{G},t-(\Delta h-1)}, \ldots, X_{\mathcal{G},t-1}\right], \tag{4}$$

$$X_d = \left[X_{\mathcal{G},t-\Delta d\cdot l_d}, X_{\mathcal{G},t-(\Delta d-1)\cdot l_d}, \ldots, X_{\mathcal{G},t-l_d}\right], \tag{5}$$

$$X_w = \left[X_{\mathcal{G},t-\Delta w\cdot l_w}, X_{\mathcal{G},t-(\Delta w-1\cdot l_w)}, \ldots, X_{\mathcal{G},t-l_w}\right], \tag{6}$$

where l_d, l_w denote the time period of a day and a week respectively.

4.2.2 Spatial attention block

We notice that important features are often concentrated in a certain region, thus a spatial attention mechanism can be introduced to focus on different regions of the feature map in space, telling the network where the region of interest is located. Besides, traffic flows are not only spatially correlated, but also have complex local spatial heterogeneity, so we propose here to use a spatial attention mechanism for traffic prediction. Specifically, we design a spatio-temporal unit module based on the attention mechanism, which captures the

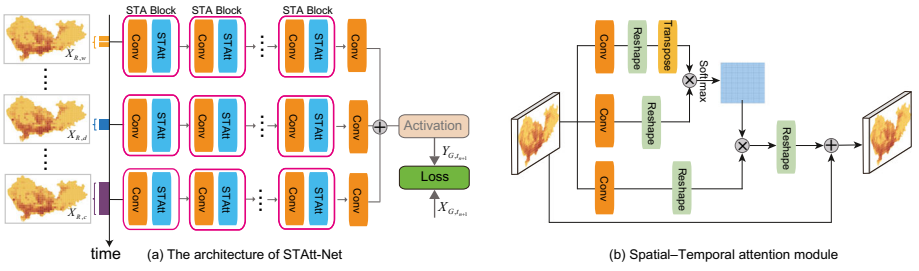

Fig. 2 Network architecture of our *STAtt-Net*, which mainly consists of three modules: (i) a *temporal dependency module* including weekly *trend*, daily *periodicity*, and hourly *closeness* components to learn periodic patterns; (ii) a *STblock module* taking advantage of Attention mechanism (b) to learn global spatio-temporal dependence(iii) a *fusion and activation module* to fuse temporal components and activate the final prediction

rich spatio-temporal relationships between regions over the whole city so as to obtain more significant spatial dependence.

Let $X_{\mathcal{G}}^{(l)} \in \mathbb{R}^{c \times i \times j}$ be the feature map extracted by the l-th ST-Block layer, where c is the number of channels and $i \times j$ is the size of the feature map. Figure 2(b) gives an illustration of the spatial attention model. CNNs are more suitable for processing data with Euclidean structure, which can better model spatial correlation. First, we improve the nonlinear representation capability of the model with a convolution operation , which can be regarded as the weighted sum of samples:

$$X_{\mathcal{G},conv}^{(l+1)} = f_c(W_{\mathcal{G}}^{(l)} * X_{\mathcal{G}}^{(l)}), \tag{7}$$

where $*$ denotes the convolution operation between a filter and the input feature maps, while $W_{\mathcal{G}}^{(l)}$ is a learnable filter in the l-th convolution layer, $f_c(\cdot)$ refers to the rectified linear unit (ReLU) *ie.* and $f_c(z) = \max(0, z)$ is the activation function.

Besides, the global and local density distributions have certain regularities due to the constant movement changes of the vehicle flows. To encode the two types of observations described above, we design a spatial attention model that is capable of modelling a large range of contextual information and capturing changes in the density distribution of crowd flows. Figure 2 gives an illustration of the spatial attention mechanism structure. The feature map $X_{\mathcal{G},conv}^{(l+1)} \in \mathbb{R}^{c \times i \times j}$ output from the previous convolution operation is fed into each of the three 1×1 convolution operations to generate three feature maps $\mathcal{F}_1$, $\mathcal{F}_2$ and $\mathcal{F}_3$, and reshape them into $\mathbb{R}^{c \times n}$, where $n = i \times j$. For $\mathcal{F}_1$, a further transpose operation is required. Next, we apply matrix multiplication and softmax operations to feature map F_1 and F_2 to obtain spatial attention maps $\mathcal{W} \in \mathbb{R}^{n \times n}$. $\mathcal{W}$ is defined as follow:

$$\mathcal{W}_{j,i} = \frac{\exp\left(\mathcal{F}_1^i \cdot \mathcal{F}_2^j\right)}{\sum_{i=1}^{n} \exp\left(\mathcal{F}_1^i \cdot \mathcal{F}_2^j\right)}, \tag{8}$$

here $\mathcal{W}_{j,i}$ represents the effect of position i on position j , a larger value means a higher similarity between position i and position j.

After generating the spatial attention matrix, we once again use the multiplication operation between $\mathcal{W}$ and $\mathcal{F}_3$ and reshape the result to $\mathbb{R}^{c \times i \times j}$. The final output of the spatial attention block is defined as:

$$X_{\mathcal{G},j}^{(l+1)} = \lambda \sum_{i=1}^{n} \left(\mathcal{W}^{j,i} \cdot \mathcal{F}_3^i\right) + X_{\mathcal{G},conv}^{(l+1),j}, \tag{9}$$

where λ is a learnable parameter. As can be seen from the detailed description of the entire model, the final output is actually a weighted sum of the features at all locations and the original features, which contain global features and selective features according to the spatial attention map.

4.2.3 Fusion and activation module

In STAtt-Net, the last layer is a fusion layer that fuses the three components to modeling spatial-temporal correlation, including closeness, daily, weekly:

$$X_{\mathcal{G}}^{Fusion} = W_h \circ X'_{\mathcal{G},h} + W_d \circ X'_{\mathcal{G},d} + W_w \circ X'_{\mathcal{G},w}, \tag{10}$$

where W_h, W_d and W_w are learnable parameters matrices and $X'_{\mathcal{G},h}$, $X'_{\mathcal{G},d}$ and $X'_{\mathcal{G},w}$ are predicted results by three components based on historical data respectively. The $\circ$ is

hadamard product which is formed by the elementwise multiplication of their elements. $X_{\mathcal{G}}^{Fusion}$ denotes the output of the merge layer. After merging the three components, we employee the active function at this phase, and the predicted value at the tth time interval is denoted by $Y_{\mathcal{R},t_{n+1}}$, the final output of STAtt-Net is then derived as:

$$Y_{\mathcal{G},t_{n+1}} = \tanh\left(X_{\mathcal{G}}^{Fusion}\right), \tag{11}$$

4.3 Training

In the end, we predict inflow and outflow simultaneously. Our model can be trained end-to-end via back-propagation by minimizing the mean square error (MSE) between the predicted traffic flow $Y_{R,t_{n+1}}$ and the ground truth $X_{\mathcal{G},t_{n+1}}$. The loss function is defined as:

$$\mathcal{L}(\theta) = \left\| Y_{\mathcal{G},t_{n+1}} - X_{\mathcal{G},t_{n+1}} \right\|_2^2, \tag{12}$$

where θ is learnable parameters in our model.

5 Experinments

5.1 Experinmental setting

All experiments were conducted in Ubuntu16.04 (64bit) with AMD Ryzen 7 2700 8-Core Processor ×16 @ 3.60GHz CPU and NVIDIA GeForce RTX 2080 Ti GPU. The STAtt-Net model is implemented under an open-sources framework Keras with TensorFlow backend. During the training phase, the model was optimized by the Adam optimizer with a learning rate of 0.0002. The batch size was set as 64. The datasets were scaled into the range [-1,1] using Mmn-max normalization. Notice that we denormalized the predicted values to compare with the true values in the evaluation phase. In order to obtain optimal model parameters and prevent overfitting, we performed the early-stopping strategy on training to control the number of epochs. All kernels of the convolutions were set to 3×3 in size. The parameters for the three temporal components were set as: $\Delta w = 1$, $\Delta d = 1$, and $\Delta c = 3$.

5.2 DataSets

We used three datasets from the real-world to assess performance of our model as mentioned before: TaxiBJ, BikeNYC, and TaxiSZ. The first two datasets are publicly available and commonly used as benchmark in various CNN based traffic prediction works [57]. While the last one, TaxiSZ, comes from our cooperation with local transportation agency. We chose it to investigate the generalization ability of our model. The statistics of the datasets are summarized in Table 2, The details are as follows:

- **TaxiBJ.** This traffic flow dataset contains 528 days GPS data of taxi in four different time periods of Beijing. After discarding corrupted data, the whole dataset is divided into 22,459 segments, with each segement to be 30-min.
- **BikeNYC.** The dataset of Bike track in New York, containing bicycle trajectory in the New York bicycle system from April 1, 2014 to September 30, 2014. Each record includes information of bicycle trip duration, trip Starting and ending time, and trip date, starting and terminal station name, station number, station longitude and latitude,

Table 2 Statistics of the datasets used in the experiments

Dataset	TaxiBJ	BikeNYC	TaxiSZ
Data type	TaxiGPS	Bike rent	TaxiGPS
Location	Beijing	NewYork	Shenzhen
Time period	528	183	181
Time interval	30 min	1 h	30 min
Grid map size	(32,32)	(16,8)	(50,25)&(100,50)
Available time interval	22,459	4,392	8,688

bicycle ID, etc. There are 183 days of records in total. The dataset is divided into 60-min segments, resulting in a total of 4392 segments with records less than 60 s excluded.

- **TaxiSZ.** The data record of taxi transactions in Shenzhen carried out by more than 20k taxis over the duration from 1st Jan. 2019 to 30th Jun. 2019. There are approximately 800k transactions record every day, leading to over 145million transactions. For each taxi transaction record, the following attributes are recorded: taxi ID, price, operating mileage, *get-on position* (denoted as m_{p0}) and time (m_{t0}), and *get-off position* (m_{p1}) and *time* (m_{t1}). The raw data contains numerously corrupted or incomplete information, such as positions outside of Shenzhen or missing get-on/get-off times. After data wiping, 128 million accurate transaction records were reserved.

5.3 Baselines

To assess the performance of our model, we compared *STAtt-Net* with the following baseline:

- **HA**: Historical average method predicts the trend of data using the average of historical mobile traffic flow in data within relatively identical time intervals of a given range.
- **ARIMA**: Autoregressive Integrated Moving Average Model is one of the classic time series forecasting models, and it was used in traffic flow prediction earlier. ARIMA regards the time series of data as a random time series, transforms the non-stationary data into a stationary series through several differences, and fits the time series into the parameter model.
- **ST-ResNet**: The residual network based model, proposed by Zheng et al. [57], can fit the traffic flow data by capturing the time correlation of traffic flow and combining with external information (date attribute and weather data, etc.).
- **ST-3DNet**: ST-3DNet uses a specially designed 3D CNN structure to learn the temporal and spatial features of traffic flow dataset together.
- **T-GCN** [55]: T-GCN combined graph convolutional network and gated recurrent units to capture the complex spatial and temporal dependencies in traffic speed prediction
- **DeFlow-Net** [53]: DeFlow-Net, a deep deformable convolutional residual network based on deformable convolutions. It is one of the most advanced convolution based deep traffic flow prediction models.

5.4 Evaluation metrics

We use Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) to evaluate our proposed network performance.They are defined as follows:

Table 3 Comparison with baseline models

Model	TaxiBJ			NYCBike			TaxiSZ		
	RMSE	MAE	MASE	RMSE	MAE	MASE	RMSE	MAE	MASE
HA	52.77	29.77	0.605	10.76	6.13	0.230	12.41	3.07	0.409
ARIMA	28.46	15.81	0.672	9.98	6.22	0.245	11.37	3.21	0.437
ST-ResNet	17.34	9.80	0.295	6.48	3.50	0.171	6.54	1.57	0.395
ST-3DNet	17.14	9.63	0.292	5.95	3.56	0.168	5.62	1.50	**0.234**
T-GCN	39.68	19.88	0.633	8.78	5.98	0.221	9.36	2.87	0.411
DeFlow-Net	**15.90**	**9.27**	**0.278**	**5.85**	3.01	0.165	**5.35**	1.50	0.236
STAtt-Net	16.64	9.39	0.281	5.95	**2.93**	**0.163**	5.41	**1.48**	0.236

$$RMSE = \sqrt{\frac{1}{N}\sum_{g=1}^{N}\left(x_{g,t_{n+1}} - y_{g,t_{n+1}}\right)^2}, \tag{13}$$

$$MAE = \frac{1}{N}\sum_{g=1}^{N} \mid x_{g,t_{n+1}} - y_{g,t_{n+1}} \mid, \tag{14}$$

where $x_{g,t_{n+1}}$ and $y_{g,t_{n+1}}$ represent the real value and the predicted value at time frame t and grid g respectively, N is the number of all all the samples for prediction. RMSE and MAE are common indices used in traffic forecasting. However, as pointed out in literature [53], RMSE measurements are unit dependent, making it unsuitable for comparison between different datasets. To address the problem, we further incorporate Mean Absolute Scaled Error (MASE), which can be express as :

$$I\left(x_{g_i,t}\right) = \frac{x_{g_i,t} - \bar{X}_{\mathcal{G},t}}{S^2} \sum_{j=1, j\neq i}^{n} w_{ij}\left(x_{g_j,t} - \bar{X}_{\mathcal{G},t}\right) \tag{15}$$

where $x_{g,t_{n+1}}$ and $y_{g,t_{n+1}}$ in the numerator are from the testing data, while $x_{g,t}$ and $x_{g,t-m}$ in the denominator are from the training data, respectively. T is the total number of time slots in the training data, and m is the seasonality of the time series (i.e., 48 for *TaxiBJ* and *TaxiSZ*, and 24 for *BikeNYC*). MASE is unit independent, allowing us to compare traffic flow predictions in different cities and at different scales. Moreover, MASE can handle actual values of zero and is not biased by very extreme values, which are problematic for mean absolute percentage error (MAPE) [14]. In general, a MASE less than 1 indicates a model is better than the naive model, and lower MASE indicates better model.

5.5 Performance comparison

The experimental results are shown in Table 3. It includes a comparison of the proposed model with the five baselines mentioned above. The best performance of all methods is marked in bold. We can observe that traditional time series methods, such as *ARIMA*,*HA*, cannot obtain good traffic forecasting results because they rely only on historical records to predict future values. Machine learning-based methods such as SVR can achieve better performance results, is limited in modeling the complex temporal and spatial dependencies in traffic forecasting. Deep learning-based methods such as *ST-ResNet*, *ST3DNet* aslo have better performance, but they are still worse than our *STAtt-Net*, which introduces attention

mechanism and multiple time components to modelling a large range of contextual information and spatio-temporal dependencies. Recently, graph-based methods are effective for the problem of traffic flow prediction. We also conducted additional comparison using a recent GNN model for traffic prediction, namely T-GCN. The results became very bad However, the spatial features learned in GCN are not optimal for the grid-based traffic network prediction. The reasons for this result is that our works aims to predict traffic flows for regions, in which CNNs are more suitable because convolutions can better model spatial correlation by decomposing the traffic network as grids. In contrast, GNN models are more appropriate for graph-structured traffic data. ***STAtt-Net*** consistently achieves the better accuracy among all the compared models with the smallest RMSE value 16.64, 5.95,5.41, MAE value 9.39, 2.93, 1.48 and MASE value 0.281, 0.163, 0.236. In general, the prediction accuracy of our method is better than all the other methods in either RMSE, MAE or MASE, except for the latest DeFlow-Net which is slightly better than ours. However, the time cost of DeFlow-Net is about four times of ours.

5.6 Comparison of different partitioning shapes and scales

Deep learning-based traffic flow prediction is affected by the plasticity area cell problem, which causes perturbations in the prediction results [52]. To explore the prediction performance of ***STAtt-Net*** on different partitioning shapes (grids *vs.* TAZs) and scales (50×25 *vs.* 100×50). The results are listed in Table 4. We can conclude that the RMSE results based on TAZ-partition are always better than grid-partition at the same scale. A potential reason is that some grids at this scale are too large and contain multiple small TAZs, resulting in the information of smaller TAZs are lost, and may even interfere with the prediction results. In addition, at scale (100×50) are records the improvements of at least 48.79% on RMSE and 54.42% on MAE at grid-partition, and 56.54% on RMSE and 66.67% on MAE at TAZ-partition compared to scale (50×25). The results infer that finer scale 100×50 is better.

5.7 Impact of the number of ST-Block layers

The number of ST-Block layers can affect the prediction result. To investigate how it can affecting *ST-ResNet* efficiency, we change the number of layers of ST-Block from 1 to 5 and the model to get different predictions. As shown in Fig. 3, the number of ST-Block layers also great affects the experinment result. Taking Fig. 3(a) as an example, when the number of ST-Block layers increases from 1 to 4, the RMSE and MAE declines continuously to 5.96 and 2.93. The same is true for the results on the *TaxiSZ*. The RMSE reduces to 5.41 and MAE reduces to 1.48. This shows that an appropriate number can improve the prediction accuracy of the network.

Table 4 Performance comparison of different partition shapes (grid *vs.* TAZ) and scales (50×25 *vs.* 100×50) on TaxiSZ

Convolution	Metric	50×25		100×50	
		Grid	TAZ	Grid	TAZ
STAtt-Net	RMSE	5.41	4.97	2.77	2.16
	MAE	1.48	1.65	0.67	0.55

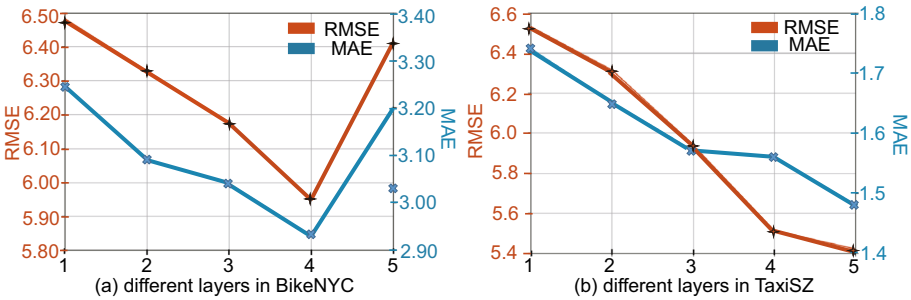

Fig. 3 Performance comparison of different numbers of *ST-Block*,the numbers of *ST-Block* layers increases from 1 to 5

6 Conclusion

In this paper, a spatial-temporal attention based convolutional networks, called STAtt-Net, is proposed for short-term traffic prediction. We developed the ST-Block module to enhance the feature extraction capabilities for learning the saptial heterogeneity. Besides, considering the temporal properties of traffic data, STAtt-Net models the temporal dependency as i) trend for weekly trend, ii) period for daily periodicity, and iii) closeness for recent time dependence. We evaluated our model on three large-scale datasets, respectively. The experimental results demonstrate that *STAtt-Net* significatly outperforms state-of-art approaches. Our method achieves a good balance between accuracy and efficiency. Since each region in STBlock has to capture global contextual information, this leads to a large computational complexity for the whole attention mechanism module. But it is still more efficient than pure 3D convolution model, such as DeFlow-Net, as revealed in the experiments. Another issue is that pure attention-based models are known to be quite 'data-hungry' as they usually require huge amounts of data to pre-train before being applicable. Finally, the interpretability of deep model is still quite challenging and the introduction of attention mechanism aggravates the issue. We note some novel work such as TFT [20], a multilayer pure deep learning model for time series with an LSTM encoder-decoder and a new attention mechanism that provides interpretable predictions. This provides us with some ideas for the next step. Besides, we will consider introduce other mechanisms such as transformers to optimize the predictive capabilities of the model in the future.

Funding This work is supported by National Natural Science Foundation of China (62077039) and Research Project (PZ2020016).

Declarations

Conflict of interest There are no other competing interests.

References

1. Atrish A, Singh N, Kumar K, Kumar V (2017) An automated hierarchical framework for player recognition in sports image. In: Proceedings of the international conference on video and image processing, pp 103–108
2. Breiman L (2001) Random forests. Mach Learn 45(1):5–32

3. Chang H, Lee Y, Yoon B, Baek S (2012) Dynamic near-term traffic flow prediction: system-oriented approach based on past experiences. IET Intell Transp Syst 6(3):292–305
4. Cheng S, Lu F, Peng P (2020) Short-term traffic forecasting by mining the non-stationarity of spatiotemporal patterns. IEEE Trans Intell Transp Syst 22(10):6365–6383
5. Choi H, Cho K, Bengio Y (2018) Fine-grained attention mechanism for neural machine translation. Neurocomputing 284:171–176
6. Deng D, Shahabi C, Demiryurek U, Zhu L, Yu R, Liu Y (2016) Latent space model for road networks to predict time-varying traffic. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp 1525–1534
7. Ding QY, Wang XF, Zhang XY, Sun ZQ (2011) Forecasting traffic volume with space-time arima model. In: Advanced materials research, vol 156. Trans Tech Publications, pp 979–983
8. Fu R, Zhang Z, Li L (2016) Using lstm and gru neural network methods for traffic flow prediction. In: 2016 31st Youth academic annual conference of Chinese association of automation (YAC). IEEE, pp 324–328
9. Guo S, Lin Y, Li S, Chen Z, Wan H (2019) Deep spatial–temporal 3d convolutional neural networks for traffic data forecasting. IEEE Trans Intell Transp Syst 20(10):3913–3926
10. Harvey AC (1990) Forecasting structural time series models and the kalman filter
11. He Z, Chow C-Y, Zhang J-D (2019) Stcnn: a spatio-temporal convolutional neural network for long-term traffic prediction. In: 2019 20th IEEE international conference on mobile data management (MDM). IEEE, pp 226–233
12. Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7132–7141
13. Huang W, Song G, Hong H, Xie K (2014) Deep architecture for traffic flow prediction: deep belief networks with multitask learning. IEEE Trans Intell Transp Syst 15(5):2191–2201
14. Hyndman RJ, Koehler AB (2006) Another look at measures of forecast accuracy. Int J Forecast 22(4):679–688
15. Ke R, Li W, Cui Z, Wang Y (2020) Two-stream multi-channel convolutional neural network for multi-lane traffic speed prediction considering traffic volume impact. Transp Res Rec 2674(4):459–470
16. Kumar K (2021) Text query based summarized event searching interface system using deep learning over cloud. Multimed Tools Appl 80(7):11079–11094
17. Kumar K, Kumar A, Bahuguna A (2017) D-cad: deep and crowded anomaly detection. In: Proceedings of the 7th international conference on computer and communication technology, pp 100–105
18. Kumar K, Shrimankar DD, Singh N (2018) Somes: an efficient som technique for event summarization in multi-view surveillance videos 383–389
19. Kumar A, Purohit K, Kumar K (2019) Stock price prediction using recurrent neural network and long short-term memory. In: International conference on deep learning, artificial intelligence and robotics. Springer, pp 153–160
20. Lim B, Arık SÖ, Loeff N, Pfister T (2021) Temporal fusion transformers for interpretable multi-horizon time series forecasting. Int J Forecast 37(4):1748–1764
21. Liu K, Gao S, Qiu P, Liu X, Yan B, Lu F (2017) Road2vec: measuring traffic interactions in urban road system from massive travel routes. ISPRS Int J Geo-Inf 6(11):321
22. Lu F, Liu K, Duan Y, Cheng S, Du F (2018) Modeling the heterogeneous traffic correlations in urban road systems using traffic-enhanced community detection approach. Physica A: Stat Mech Appl 501:227–237
23. Luo Q, Zhou Y (2021) Spatial-temporal structures of deep learning models for traffic flow forecasting: a survey. In: 2021 4th International conference on intelligent autonomous systems (ICoIAS). IEEE, pp 187–193
24. Luong M-T, Pham H, Manning CD (2015) Effective approaches to attention-based neural machine translation. arXiv:1508.04025
25. 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. Transp Res Part C: Emerg Technol 54:187–197
26. Martínez LM, Viegas JM, Silva EA (2009) A traffic analysis zone definition: a new methodology and algorithm. Transportation 36(5):581–599
27. Mnih V, Heess N, Graves A (2014) Recurrent models of visual attention. In: Advances in neural information processing systems, pp 2204–2212
28. Moorthy C, Ratcliffe B (1988) Short term traffic forecasting using time series methods. Transp Plan Technol 12(1):45–56
29. Negi A, Kumar K (2021) Face mask detection in real-time video stream using deep learning. In: Computational intelligence and healthcare informatics, pp 255–268
30. Negi A, Kumar K (2022) Chapter 1—ai-based implementation of decisive technology for prevention and fight with covid-19 1–14

31. Negi A, Kumar K, Chaudhari NS, Singh N, Chauhan P (2021) Predictive analytics for recognizing human activities using residual network and fine-tuning. In: International conference on big data analytics. Springer, pp 296–310
32. Okutani I, Stephanedes YJ (1984) Dynamic prediction of traffic volume through kalman filtering theory. Transp Res Part B: Methodol 18(1):1–11
33. Openshaw S (1984) The modifiable areal unit problem. Geo Books, Norwick
34. Sharma S, Kumar K (2021) Asl-3dcnn: American sign language recognition technique using 3-d convolutional neural networks. Multimed Tools Appl 80(17):26319–26331
35. Shi X, Chen Z, Wang H, Yeung D-Y, Wong W-K, Woo W-C (2015) Convolutional lstm network: a machine learning approach for precipitation nowcasting. arXiv:1506.04214
36. Smola AJ, Schölkopf B (2004) A tutorial on support vector regression. Stat Comput 14(3):199–222
37. Solanki A, Bamrara R, Kumar K, Singh N (2020) VEDL: a novel video event searching technique using deep learning. In: Soft computing: theories and applications, pp 905–914
38. Srinivasu PN, Balas VE (2021) Self-learning network-based segmentation for real-time brain mr images through haris. PeerJ Comput Sci 7:654
39. Stathopoulos A, Karlaftis M (2001) Temporal and spatial variations of real-time traffic data in urban areas. Transp Res Rec 1768(1):135–140
40. Tang C, Zhu X, Liu X, Wang L, Zomaya A (2019) Defusionnet: defocus blur detection via recurrently fusing and refining multi-scale deep features. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 2700–2709
41. Tang C, Liu X, An S, Wang P (2021) Br^2net: defocus blur detection via a bidirectional channel attention residual refining network. IEEE Trans Multimed 23:624–635. https://doi.org/10.1109/TMM.2020.2985541
42. Tobler WR (1970) A computer movie simulating urban growth in the detroit region. Econ Geogr 46(2)
43. Wang X, Girshick R, Gupta A, He K (2018) Non-local neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7794–7803
44. Williams BM, Hoel LA (2003) Modeling and forecasting vehicular traffic flow as a seasonal arima process: theoretical basis and empirical results. J Transp Eng 129(6):664–672
45. Williams BM, Durvasula PK, Brown DE (1998) Urban freeway traffic flow prediction: application of seasonal autoregressive integrated moving average and exponential smoothing models. Transp Res Rec 1644(1):132–141
46. Xingjian S, Chen Z, Wang H, Yeung D-Y, Wong W-K, Woo W-C (2015) Convolutional lstm network: a machine learning approach for precipitation nowcasting. In: Advances in neural information processing systems, pp 802–810
47. Yan Q, Wang B, Zhang W, Luo C, Xu W, Xu Z, Zhang Y, Shi Q, Zhang L, You Z (2020) Attention-guided deep neural network with multi-scale feature fusion for liver vessel segmentation. IEEE J Biomed Health Inform 25(7):2629–2642
48. Yan Q, Wang B, Zhang W, Luo C, Xu W, Xu Z, Zhang Y, Shi Q, Zhang L, You Z (2021) Attention-guided deep neural network with multi-scale feature fusion for liver vessel segmentation. IEEE J Biomed Health Inform 25(7):2629–2642. https://doi.org/10.1109/JBHI.2020.3042069
49. Yao Z-S, Shao C-F, Gao Y-L (2006) Research on methods of short-term traffic forecasting based on support vector regression [j]. J Beijing Jiaotong Univ 30(3):19–22
50. Yao H, Tang X, Wei H, Zheng G, Li Z (2019) Revisiting spatial-temporal similarity: a deep learning framework for traffic prediction. In: Proceedings of the AAAI conference on artificial intelligence, vol 33, pp 5668–5675
51. Yue Y (2006) Spatial-temporal dependency of traffic flow and its implications for short-term traffic forecasting. HKU Theses Online (HKUTO)
52. Zeng W, Lin C, Lin J, Jiang J, Xia J, Turkay C, Chen W (2020) Revisiting the modifiable areal unit problem in deep traffic prediction with visual analytics. IEEE Trans Vis Comput Graph 27(2):839–848
53. Zeng W, Lin C, Liu K, Lin J, Tung AK (2021) Modeling spatial nonstationarity via deformable convolutions for deep traffic flow prediction. IEEE Trans Knowl Data Eng
54. Zeng W, Lin C, Lin J, Jiang J, Xia J, Turkay C, Chen W (2021) Revisiting the modifiable areal unit problem in deep traffic prediction with visual analytics. IEEE Trans Visual Comput Graph 27(2):839–848. https://doi.org/10.1109/TVCG.2020.3030410
55. Zhao L, Song Y, Zhang C, Liu Y, Wang P, Lin T, Deng M, Li H (2019) T-gcn: a temporal graph convolutional network for traffic prediction. IEEE Trans Intell Transp Syst 21(9):3848–3858
56. Zhang X-L, He G-G, Lu H-P (2009) Short-term traffic flow forecasting based on k-nearest neighbors non-parametric regression. J Syst Eng 24(2):178–183
57. Zhang J, Zheng Y, Qi D (2017) Deep spatio-temporal residual networks for citywide crowd flows prediction. In: Proceedings of the AAAI conference on artificial intelligence, vol 31

58. Zheng C, Fan X, Wang C, Qi J (2020) Gman: a graph multi-attention network for traffic prediction. In: Proceedings of the AAAI conference on artificial intelligence, vol 34, pp 1234–1241