A3T-GCN: Attention Temporal Graph Convolutional Network for Traffic Forecasting

Jiawei Zhu, Yujiao Song, Lin Zhao and Haifeng Li*

Abstract—Accurate real-time traffic forecasting is a core technological problem against the implementation of the intelligent transportation system. However, it remains challenging considering the complex spatial and temporal dependencies among traffic flows. In the spatial dimension, due to the connectivity of the road network, the traffic flows between linked roads are closely related. In terms of the temporal factor, although there exists a tendency among adjacent time points in general, the importance of distant past points is not necessarily smaller than that of recent past points since traffic flows are also affected by external factors. In this study, an attention temporal graph convolutional network (A3T-GCN) traffic forecasting method was proposed to simultaneously capture global temporal dynamics and spatial correlations. The A3T-GCN model learns the short-time trend in time series by using the gated recurrent units and learns the spatial dependence based on the topology of the road network through the graph convolutional network. Moreover, the attention mechanism was introduced to adjust the importance of different time points and assemble global temporal information to improve prediction accuracy. Experimental results in real-world datasets demonstrate the effectiveness and robustness of proposed A3T-GCN. The source code can be visited at https://github.com/lehaifeng/T-GCN/A3T.

Index Terms—traffic forecasting, attention temporal graph convolutional network, spatial dependence, temporal dependence

1 Introduction

Traffic forecasting is an important component of intelligent transportation systems and a vital part of transportation planning and management and traffic control [**?** **?** **?** **?**]. Accurate real-time traffic forecasting has been a great challenge because of complex spatiotemporal dependencies. Temporal dependence means that traffic state changes with time, which is manifested by periodicity and tendency. Spatial dependence means that changes in traffic state are subject to the structural topology of road networks, which is manifested by the transmission of upstream traffic state to downstream sections and the retrospective effects of downstream traffic state on the upstream section[**?**]. Hence, considering the complex temporal features and the topological characteristics of the road network is essential in realizing the traffic forecasting task.

Existing traffic forecasting models can be divided into parametric and non-parametric models. Common parametric models include historical average, time series [**?** **?**], linear regression [**?**], and Kalman filtering models[**?**]. Although traditional parametric models use simple algorithms, they depend on stationary hypothesis. These models can neither reflect nonlinearity and uncertainty of traffic states nor overcome the interference of random events, such as traffic accidents. Non-parametric models can solve these problems well because they can learn the statistical laws of data automatically with adequate historical data. Common non-parametric models include k-nearest [**?**], support vector regression (SVR) [**?** **?**], fuzzy logic [**?**], Bayesian network[**?**], and neural network models.

Recently, deep neural network models have attracted wide attention from scholars because of the rapid development of deep learning [**?** **?**]. Recurrent neural networks (RNNs), long short-term memory (LSTM) [**?**], and gated recurrent units (GRUs)[**?**] have been successfully utilized in traffic forecasting because they can use self-circulation mechanism and model temporal dependence [**?** **?**]. However, these models only consider the temporal variation of traffic state and neglect spatial dependence. Many scholars have introduced convolutional neural networks (CNNs) in their models to characterize spatial dependence remarkably. Wu et al. [**?**] designed a feature fusion framework for short-term traffic flow forecasting by combining a CNN with LSTM. The framework captured the spatial characteristics of traffic flow through a one-dimensional CNN and explored short-term variations and periodicity of traffic flow with two LSTMs. Cao et al. [**?**] proposed an end-to-end model called ITRCN, which transformed the interactive network flow to images and captured network flows using a CNN. ITRCN also extracted temporal features by using GRU. An experiment proved that the forecasting error of this method was 14.3% and 13.0% higher than those of GRU and CNN, respectively. Yu et al. [**?**] captured spatial correlation and temporal dynamics by using DCNN and LSTM, respectively. They also proved the superiority of SRCN based on the investigation on the traffic network data in Beijing.

Although CNN is actually applicable to Euclidean data [**?**], such as image and grids, it still has limitations in traffic networks, which possess non-Euclidean structures. In recent years, graph convolutional network (GCN) [**?**], which can overcome the abovementioned limitations and capture structural characteristics of networks, has rapidly developed [**?** **?** **?**]. In addition, RNNs and their variants use sequential processing over time and more apt to remember the latest information, thus are suitable to capture evolving short-term tendencies. While The importance of different time points cannot be distinguished only by the proximity of time. Mechanisms that are capable of learning global

• *H. Li, J. Zhu, Y. Song and L. Zhao are with School of Geosciences and Info-Physics, Central South University, Changsha 410083, China.*

correlations are needed.

For this reason, an attention temporal GCN (A3T-GCN) was proposed for traffic forecasting task. The A3T-GCN combines GCNs and GRUs and introduces an attention mechanism[**?** **?**]. It not only can capture spatiotemporal dependencies but also ajust and assemble global variation information. The A3T-GCN is used for traffic forecasting on the basis of urban road networks.

2 A3T-GCN

2.1 Definition of problems

In this study, traffic forecasting is performed to predict future traffic state according to historical traffic states on urban roads. Generally, traffic state can refer to traffic flow, speed, and density. In this study, traffic state only refers to traffic speed.

Definition 1. Road network G: The topological structure of urban road network is described as $G = (V, E)$,where $V = \{v_1, v_2, \cdots, v_N\}$ is the set of road section, and N is the number of road sections. E is the set of edges, which reflects the connections between road sections. The whole connectivity information is stored in the adjacent matrix $A \in R^{N \times N}$, where rows and columns are indexed by road sections, and the value of each entry indicates the connectivity between corresponding road sections. The entry value is 0 if there is no existed link between roads and 1 (unweighted graph) or non-negative (weighted graph) if otherwise.

Definition 2. Feature matrix $X^{N \times P}$: Traffic speed on a road section is viewed as the attribute of network nodes, and it is expressed by the feature matrix $X \in R^{N \times P}$, where P is the number of node attribute features, that is, the length of historical time series. X_i denotes the traffic speed in all sections at time i.

Therefore, the traffic forecasting modelling temporal and spatial dependencies can be viewed as learning a mapping function f on the basis of the road network G and feature matrix X of the road network. Traffic speeds of future T moments are calculated as follows:

$$[X_{t+1}, \cdots, X_{t+T}] = f\left(G; (X_{t-n}, \cdots, X_{t-1}, X_t)\right) \quad (1)$$

where n is the length of a given historical time series, and T is the length of time series that needs to be forecasted.

2.2 GCN model

GCNs are semi-supervised models that can process graph structures. They are an advancement of CNNs in graph fields. GCNs have achieved many progresses in many applications, such as image classification [**?**], document classification [**?**], and unsupervised learning [**?**]. Convolutional mode in GCNs includes spectrum and spatial domain convolutions [**?**]. The former was applied in this study. Spectrum convolution can be defined as the product of signal x on the graph and figure filter $g_\theta(L)$,which is constructed in the Fourier domain:$g_\theta(L) * x = U g_\theta(U^T x)$, where θ is a model parameter, L is the graph Laplacian matrix, U is the eigenvector of normalized Laplacian matrix $L = I_N - D^{-\frac{1}{2}} A D^{-\frac{1}{2}} = U \lambda U^T$, and $U^T x$ is the graph Fourier transformation of x. x can also be promoted to $X \in R^{N \times C}$, where C refers to the number of features.

Given the characteristic matrix X and adjacent matrix A, GCNs can replace the convolutional operation in anterior CNNs by performing the spectrum convolutional operation with consideration to the graph node and first-order adjacent domains of nodes to capture the spatial characteristics of graph. Moreover, hierarchical propagation rule is applied to superpose multiple networks. A multilayer GCN model can be expressed as:

$$H^{(l+1)} = \sigma\left(\widetilde{D}^{-\frac{1}{2}} \widehat{A} \widetilde{D}^{-\frac{1}{2}} H^{(l)} \theta^{(l)}\right) \quad (2)$$

where $\widetilde{A} = A + I_N$ is an adjacent matrix with self-connection structures, I_N is an identity matrix, $\widetilde{D}$ is a degree matrix, $\widetilde{D}_{ii} = \sum_j \widetilde{A}_{ij}$, $H^{(l)} \in R^{N \times l}$ is the output of layer l, $\theta^{(l)}$ is the parameter of layer l, and $\sigma(\cdot)$ is an activation function used for nonlinear modeling.

Generally, a two-layer GCN model [**?**] can be expressed as:

$$f(X, A) = \sigma\left(\widehat{A}\, ReLU\left(\widehat{A}\, X\, W_0\right) W_1\right) \quad (3)$$

where X is a feature matrix; A is the adjacent matrix; and $\widehat{A} = \widetilde{D}^{-\frac{1}{2}} \widetilde{A} \widetilde{D}^{-\frac{1}{2}}$is a preprocessing step, where $\widetilde{A} = A + I_N$ is the adjacent matrix of graph G with self-connection structure. $W_0 \in R^{P \times H}$ is the weight matrix from the input layer to the hidden unit layer, where P is the length of time, and H is the number of hidden units. $W_1 \in R^{H \times T}$ is the weight matrix from the hidden layer to the output layer. $f(X, A) \in R^{N \times T}$ denotes the output with a forecasting length of T, and $ReLU()$is a common nonlinear activation function.

GCNs can encode the topological structures of road networks and the attributes of road sections simultaneously by determining the topological relationship between the central road section and the surrounding road sections. Spatial dependence can be captured on this basis. In a word, this study learned spatial dependence through the GCN model [**?**].

2.3 GRU model

Temporal dependence of traffic state is another key problem that hinders traffic forecasting. RNNs are neural network models that process sequential data. However, limitations in long-term forecasting are observed in traditional RNNs because of disadvantages in gradient disappearance and explosion [**?**]. LSTM [**?**] and GRUs [**?**] are variants of RNNs that mediate the problems effectively. LSTM and GRUs basically have the same fundamental principles. Both models use gated mechanisms to maintain long-term information and perform similarly in various tasks [**?**]. However, LSTM is more complicated, and it takes longer training time than GRUs, whereas GRU has a relatively simpler structure, fewer parameters, and faster training ability compared with LSTM.

In the present model, temporal dependence was captured by a GRU model. The calculation process is introduced as follows, where h_{t-1} is the hidden state at t-1, x_t is the traffic speed at the current moment, and r_t is the reset gate to control the degree of neglecting the state information at the previous moment. Information unrelated with forecasting can be abandoned. If the reset gate outputs 0, then

the traffic information at the previous moment is neglected. If the reset gate outputs 1, then the traffic information at the previous moment is brought into the next moment completely. u_t is the update gate and is used to control the state information quantity at the previous moment that is brought into the current state. Meanwhile, c_t is the memory content stored at the current moment, and h_t is the output state at the current moment.

$$u_t = \sigma(W_u * [X_t, h_{t-1}] + b_u) \tag{4}$$

$$r_t = \sigma(W_r * [X_t, h_{t-1}] + b_r) \tag{5}$$

$$c_t = \tanh(W_c [X_t, (r_t * h_{t-1})] + b_c) \tag{6}$$

$$h_t = u_t * h_{t-1} + (1 - u_t) * c_t \tag{7}$$

GRUs determine traffic state at the current moment by using hidden state at previous moment and traffic information at current moment as input. GRUs retain the variation trends of historical traffic information when capturing traffic information at current moment because of the gated mechanism. Hence, this model can capture dynamic temporal variation features from the traffic data, that is, this study has applied a GRU model to learn the temporal variation trends of the traffic state.

2.4 Attention model

Attention model is realized on the basis of encoder–decoder model. This model is initially used in neural machine translation tasks[**?**]. Nowadays, attention models are widely applied in image caption generation [**?**], recommendation system [**?**], and document classification [**?**]. With the rapid development of such models, existing attention models can be divided into multiple types, such as soft and hard attention[**?**], global and local attention[**?**], and self-attention[**?**]. In the current study, a soft attention model was used to learn the importance of traffic information at every moment, and then a context vector that could express the global variation trends of traffic state was calculated for future traffic forecasting tasks.

Suppose that a time series $x_i(i = 1, 2, \cdots, n)$,where n is the time series length, is introduced. The design process of soft attention models is introduced as follows. First, the hidden states $h_i(i = 1, 2, \cdots, n)$ at different moments are calculated using CNNs (and their variants) or RNNs (and their variant), and they are expressed as $H = \{h_1, h_2, \cdots, h_n\}$.Second, a scoring function is designed to calculate the score/weight of each hidden state. Third, an attention function is designed to calculate the context vector (Ct) that can describe global traffic variation information. Finally, the final output results are obtained using the context vector. In the present study, these steps were followed in the design process, but a multilayer perception was applied as the scoring function instead.

Particularly, the characteristics (h_i) at each moment were used as input when calculating the weight of each hidden state based on f. The corresponding outputs could be gained through two hidden layers. The weights of each characteristic (α_i) are calculated by a Softmax normalized index function (eq. (8)), where $w_{(1)}$ and $b_{(1)}$ are the weight and deviation of the first layer and $w_{(2)}$ and $b_{(2)}$ are the weight and deviation of the second layer, respectively.

$$e_i = w_{(2)}(w_{(1)}H + b_{(1)}) + b_{(2)} \tag{8}$$

$$\alpha_i = \frac{\exp(e_i)}{\sum_{k=1}^{n} \exp(e_k)} \tag{9}$$

Finally, the attention function was designed. The calculation process of the context vector (C_t) that covers global traffic variation information is shown in Equation (10).

$$C_t = \sum_{i=1}^{n} \alpha_i * h_i \tag{10}$$

2.5 A3T-GCN model

The A3t-GCN is a improvement of our previous work named T-GCN[**?**]. The attention mechanism was introduced to re-weight the influence of historical traffic states and thus to capture the global variation trends of traffic state. The model structure is shown in Fig. **??**.

A temporal GCN (T-GCN) model was constructed by combining GCN and GRU. n historical time series traffic data were inputted into the T-GCN model to obtain n hidden states (h) that covered spatiotemporal characteristics:$\{h_{t-n}, \cdots, h_{t-1}, h_t\}$.The calculation of the T-GCN is shown in eq. (11), where h_{t-1} is the output at t-1. GC is the graph convolutional process. u_t and r_t are the update and reset gates at t, respectively. c_t is the stored content at the current moment. h_t is the output state at moment t, and W and b are the weight and the deviation in the training process, respectively.

$$u_t = \sigma(W_u * [GC(A, X_t), h_{t-1}] + b_u) \tag{11}$$

$$r_t = \sigma(W_r * [GC(A, X_t), h_{t-1}] + b_r) \tag{12}$$

$$c_t = \tanh(W_c * [GC(A, X_t), (r_t * h_{t-1})] + b_c) \tag{13}$$

$$h_t = u_t * h_{t-1} + (1 - u_t) * c_t) \tag{14}$$

Then, the hidden states were inputted into the attention model to determine the context vector that covers the global traffic variation information. Particularly, the weight of each h was calculated by Softmax using a multilayer perception:$\{a_{t-n}, \cdots, a_{t-1}, a_t\}$.The context vector that covers global traffic variation information is calculated by the weighted sum. Finally, forecasting results were outputted using the fully connected layer.

In sum, we proposed the A3T-GCN to realize traffic forecasting. The urban road network was constructed into a graph network, and the traffic state on different sections was described as node attributes. The topological characteristics of the road network were captured by a GCN to obtain spatial dependence. The dynamic variation of node attributes was captured by a GRU to obtain the local temporal tendency of traffic state. The global variation trend of the traffic state was then captured by the attention model, which was conducive in realizing accurate traffic forecasting.

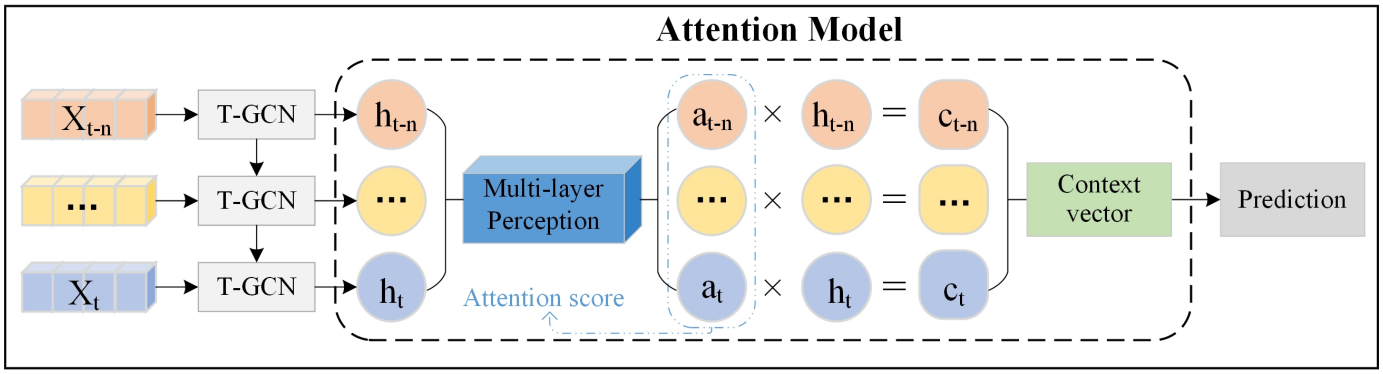


Fig. 1. A3T-GCN framework.

2.6 Loss function

Training aims to minimize errors between real and predicted speed in the road network . Real and predicted speed on different sections at t are expressed by Y and $\widehat{Y}$, respectively. Therefore, the objective function of A3T-GCN is shown as follows. The first term aims to minimize the error between real and predicted speed. The second term L_{reg} is a normalization term, which is conducive to avoid overfitting. λ is a hyper-parameter.

$$loss = \| Y_t - \widehat{Y}_t \| + \lambda L_{reg} \tag{15}$$

3 Experiments

3.1 Data Description

Two real-world traffic datasets, namely, taxi trajectory dataset (SZ_taxi) in Shenzhen City and loop detector dataset (Los_loop) in Los Angeles, were used. Both datasets are related with traffic speed. Hence, traffic speed is viewed as the traffic information in the experiments. SZ_taxi dataset is the taxi trajectory of Shenzhen from Jan. 1 to Jan. 31, 2015. In the present study, 156 major roads of Luohu District were selected as the study area. Los_loop dataset is collected in the highway of Los Angeles County in real time by loop detectors. A total of 207 sensors along with their traffic speed from Mar. 1 to Mar. 7, 2012 were selected.

3.2 Evaluation Metrics

To evaluate the prediction performance of the model, the error between real traffic speed and predicted results is evaluated on the basis of the following metrics:

(1) Root Mean Squared Error (RMSE):

$$RMSE = \sqrt{\frac{1}{MN}\sum_{j=1}^{M}\sum_{i=1}^{N}(y_i^j - \widehat{y_i^j})^2} \tag{16}$$

(2) Mean Absolute Error (MAE):

$$MAE = \frac{1}{MN}\sum_{j=1}^{M}\sum_{i=1}^{N}\left|y_i^j - \widehat{y_i^j}\right| \tag{17}$$

(3) Accuracy:

$$Accuracy = 1 - \frac{\| Y - \widehat{Y} \|_F}{\| Y \|_F} \tag{18}$$

(4) Coefficient of Determination (R^2):

$$R^2 = 1 - \frac{\sum_{j=1}^{M}\sum_{i=1}^{N}(y_i^j - \widehat{y_i^j})^2}{\sum_{j=1}^{M}\sum_{i=1}^{N}(y_i^j - \bar{Y})^2} \tag{19}$$

(5) Explained Variance Score (var):

$$var = 1 - \frac{Var\left\{Y - \widehat{Y}\right\}}{Var\{Y\}} \tag{20}$$

where y_i^j and $\widehat{y_i^j}$ are the real and predicted traffic information of temporal sample j on road i, respectively. N is the number of nodes on road. M is the number of temporal samples. Y and $\widehat{Y}$ are the set of y_i^j and $\widehat{y_i^j}$ respectively, and $\bar{Y}$ is the mean of Y.

Particularly, RMSE and MAE are used to measure prediction error. Small RMSE and MASE values reflect high prediction precision. Accuracy is used to measure forecasting precision, and high accuracy value is preferred. R^2 and var calculate the correlation coefficient, which measures the ability of the prediction result to represent the actual data: the larger the value is, the better the prediction effect is.

3.3 Experimental result analysis

The hyper-parameters of A3T-GCN include learning rate, epoch, and number of hidden units. In the experiment, learning rate and epoch were manually set on the basis of experiences as 0.001 and 5000 for both datasets. As for the number of hidden units, we set it to 64 and 100 for SZ_taxi and Los_loop, respectively.

In the present study, 80% of the traffic data are used as the training set, and the remaining 20% of the data are used as the test set. The traffic information in the next 15, 30, 45, and 60 min is predicted. The predicted results are compared with results from the historical average model (HA), auto-regressive integrated moving average model (ARIMA), SVR, GCN model, and GRU model. The A3T-GCN is analyzed from perspectives of precision, spatiotemporal prediction capabilities, long-term prediction capability, and global feature capturing capability.

(1) High prediction precision. Table **??** shows the comparisons of different models and two real datasets in terms of the prediction precision of various traffic speed lengths. The prediction precision of neural network models (e.g.,

TABLE 1
The prediction results of the T-GCN model and other baseline methods on SZ-taxi and Los-loop datasets.

T	Metric	SZ-taxi						Los-loop					
		HA	ARIMA	SVR	GCN	GRU	AT-GCN	HA	ARIMA	SVR	GCN	GRU	AT-GCN
15min	$RMSE$	4.2951	7.2406	4.1455	5.6596	3.9994	**3.8989**	7.4427	10.0439	6.0084	7.7922	5.2182	**5.0904**
	MAE	2.7815	4.9824	2.6233	4.2367	**2.5955**	2.6840	4.0145	7.6832	3.7285	5.3525	**3.0602**	3.1365
	$Accuracy$	0.7008	0.4463	0.7112	0.6107	0.7249	**0.7318**	0.8733	0.8275	0.8977	0.8673	0.9109	**0.9133**
	R^2	0.8307	$*$	0.8423	0.6654	0.8329	**0.8512**	0.7121	0.0025	0.8123	0.6843	0.8576	**0.8653**
	var	0.8307	0.0035	0.8424	0.6655	0.8329	**0.8512**	0.7121	$*$	0.8146	0.6844	0.8577	**0.8653**
30min	$RMSE$	4.2951	6.7899	4.1628	5.6918	4.0942	**3.9228**	7.4427	9.3450	6.9588	8.3353	6.2802	**5.9974**
	MAE	2.7815	4.6765	**2.6875**	4.2647	2.6906	2.7038	4.0145	7.6891	3.7248	5.6118	**3.6505**	3.6610
	$Accuracy$	0.7008	0.3845	0.7100	0.6085	0.7184	**0.7302**	0.8733	0.8275	0.8815	0.8581	0.8931	**0.8979**
	R^2	0.8307	$*$	0.8410	0.6616	0.8249	**0.8493**	0.7121	0.0031	0.7492	0.6402	0.7957	**0.8137**
	var	0.8307	0.0081	0.8413	0.6617	0.8250	**0.8493**	0.7121	$*$	0.7523	0.6404	0.7958	**0.8137**
45min	$RMSE$	4.2951	6.7852	4.1885	5.7142	4.1534	**3.9461**	7.4427	10.0508	7.7504	8.8036	7.0343	**6.6840**
	MAE	2.7815	4.6734	**2.7359**	4.2844	2.7743	2.7261	4.0145	7.6924	4.1288	5.9534	**4.0915**	4.1712
	$Accuracy$	0.7008	0.3847	0.7082	0.6069	0.7143	**0.7286**	0.8733	0.8273	0.8680	0.8500	0.8801	**0.8861**
	R^2	0.8307	$*$	0.8391	0.6589	0.8198	**0.8474**	0.7121	$*$	0.6899	0.5999	0.7446	**0.7694**
	var	0.8307	0.0087	0.8397	0.6590	0.8199	**0.8474**	0.7121	0.0035	0.6947	0.6001	0.7451	**0.7705**
60min	$RMSE$	4.2951	6.7708	4.2156	5.7361	4.0747	**3.9707**	7.4427	10.0538	8.4388	9.2657	7.6621	**7.0990**
	MAE	2.7815	4.6655	2.7751	4.3034	**2.7712**	2.7391	4.0145	7.6952	**4.5036**	6.2892	4.5186	4.2343
	$Accuracy$	0.7008	0.3851	0.7063	0.6054	0.7197	**0.7269**	0.8733	0.8273	0.8562	0.8421	0.8694	**0.8790**
	R^2	0.8307	$*$	0.8370	0.6564	0.8266	**0.8454**	0.7121	$*$	0.6336	0.5583	0.6980	**0.7407**
	var	0.8307	0.0111	0.8379	0.6564	0.8267	**0.8454**	0.7121	0.0036	0.5593	0.5593	0.6984	**0.7415**

A3T-GCN and GRU) is higher than those of other models (e.g., HA, ARIMA, and SVR). With respect to 15-minute time series, the RMSE and accuracy of HA are approximately 9.22% higher and 4.24% lower than those of A3T-GCN, respectively. The RMSE and accuracy of ARIMA are approximately 46.15% higher and 39.01% lower than those of A3T-GCN, respectively. The RMSE and accuracy of SVR are approximately 5.95% higher and 2.81% lower than those of A3T-GCN, respectively. Compared with GRU, The RMSE and accuracy of HA is approximately 6.88% higher and 3.32% lower than those of GRU, respectively. The RMSE and accuracy of ARIMA are approximately 44.76% and 38.07%, respectively. The RMSE and accuracy of SVAR are approximately 3.52% and 1.87%, respectively. These results are mainly caused by the poor nonlinear fitting abilities of HA, ARIMA, and SVAR to complicated changing traffic data. Processing long-term non-stationary data is difficult when ARIMA is used. Moreover, ARIMA is gained by averaging the errors of different sections. The data of some sections might greatly fluctuate to increase the final error. Hence, ARIMA shows the lowest forecasting accuracy.

Similar conclusions could be drawn for Los_loop. In a word, A3T-GCN model can obtain the optimal prediction performance of all metrics in two real datasets, thereby proving the validity and superiority of A3T-GCN model in spatiotemporal traffic forecasting tasks.

(2) Effectiveness of modelling both spatial and temporal dependencies. To test the benefits brought by depicting the spatiotemporal characteristics of traffic data simultaneously in A3T-GCN, the model is compared with GCN and GRU.

Fig. **??** shows the results based on SZ_taxi. Compared with GCN (considering spatial characteristics only), A3T-GCN achieves approximately 31.11%, 31.08%, 30.94%, and 30.78% lower RMSEs in 15, 30, 45, and 60 minutes of traffic forecasting time series, respectively. In sum, the prediction error of A3T-GCN is kept lower than that of GCN in 15, 30, 45, and 60 minutes of traffic forecasting. Therefore, the A3T-GCN can capture spatial characteristics.

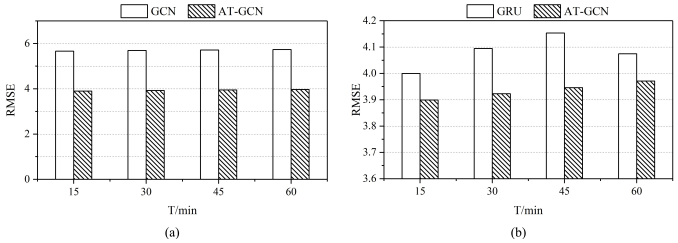


Fig. 2. SZ-taxi: Spatiotemporal prediction capabilities.

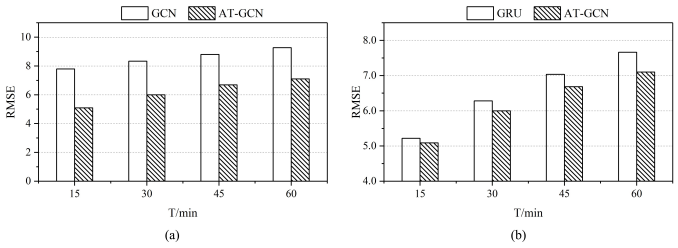


Fig. 3. Los-loop: Spatiotemporal prediction capabilities.

Compared with GRU (considering temporal characteristics only), A3T-GCN achieves approximately 2.51% lower RMSE in 15 minutes traffic forecasting, approximately 4.19% lower RMSE in 30 minutes traffic forecasting, approximately 4.99% lower RMSE in 45 minutes time series, and approximately 2.55% lower RMSE in 60 minutes time series. In sum, the prediction error of A3T-GCN is kept lower than that of GRU in 15, 30, 45, and 60 minutes traffic forecasting. Therefore, the A3T-GCN can capture temporal dependence.

Results based on Los_loop, which are similar with those based on SZ_taxi, are shown in Fig. **??**. In short, the A3T-GCN has good spatiotemporal prediction capabilities. In

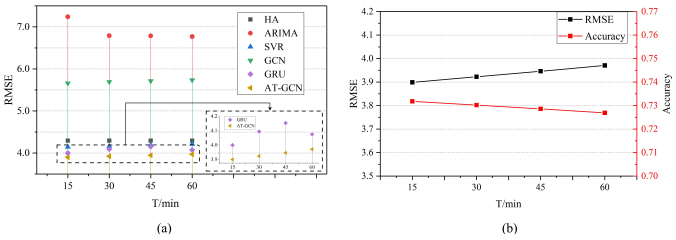


Fig. 4. SZ-taxi: Long-term prediction capability.

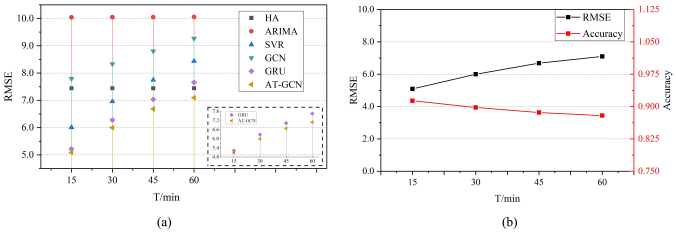


Fig. 5. Los-loop: Long-term prediction capability.

other words, A3T-GCN model can capture the spatial topological characteristics of urban road networks and the temporal variation characteristics of traffic state simultaneously.

(3) Long-term prediction capability. Long-term prediction capability of A3T-GCN was tested by traffic speed forecasting in 15, 30, 45, and 60 minutes prediction horizon. Forecasting results based on SZ-taxi are shown in Fig. **??**. The RMSE comparison of different models under different lengths of time series is shown in Fig. **??**(a). The RMSE of the A3T-GCN is the lowest under all lengths of time series. The variation trends of RMSE and accuracy, which reflects prediction error and precision, respectively, of the A3T-GCN under different lengths of time series are shown in Fig. **??**(b). RMSE increases as the length of time series increases, whereas accuracy declines slightly and shows certain stationary.

The forecasting results based on Los_loop are shown in Fig. **??**, and consistent laws are found. In sum, A3T-GCN has good long-term prediction capability. It can obtain high accuracy by training for 15, 30, 45, and 60 minutes prediction horizon. Forecasting results of A3T-GCN change slightly with changes in length of time series, thereby showing certain stationary. Therefore, the A3T-GCN is applicable to short-term and long-term traffic forecasting tasks.

(4) Effectiveness of introducing attention to capture global variation. A3T-GCN and T-GCN were compared to test the superiority of capturing global variation. Results are shown in Table **??**. A3T-GCN model shows approximately 0.86% lower RMSE and approximately 0.32% higher accuracy than T-GCN model under 15 minutes time series, approximately 1.31% lower RMSE and approximately 0.48% higher accuracy under 30 minutes time series, approximately 1.14% lower RMSE and approximately 0.43% higher accuracy under 45 minutes traffic forecasting, and approximately 0.99% lower RMSE and approximately 0.37% higher accuracy under 60 minutes time series.

Hence, the prediction error of A3T-GCN is lower than that of T-GCN, but the accuracy of the former is higher under different horizons of traffic forecasting, thereby proving the global feature capturing capability of the A3T-GCN model.

TABLE 2
Comparison of forecasting results between A3T-GCN and T-GCN under different lengths of time series based on SZ-taxi and Los-loop.

T	Metric	SZ-taxi		Los-loop	
		T-GCN	AT-GCN	T-GCN	AT-GCN
15min	$RMSE$	3.9325	**3.8989**	5.1264	**5.0904**
	MAE	2.7145	**2.6840**	3.1802	**3.1365**
	$Accuracy$	0.7295	**0.7318**	0.9127	**0.9133**
	R^2	0.8539	**0.8512**	0.8634	**0.8653**
	var	0.8539	**0.8512**	0.8634	**0.8653**
30min	$RMSE$	3.9740	**3.9228**	6.0598	**5.9974**
	MAE	2.7522	**2.7038**	3.7466	**3.6610**
	$Accuracy$	0.7267	**0.7302**	0.8968	**0.8979**
	R^2	0.8451	**0.8493**	0.8098	**0.8137**
	var	0.8451	**0.8493**	0.8100	**0.8137**
45min	$RMSE$	3.9910	**3.9461**	6.7065	**6.684**
	MAE	2.7645	**2.7261**	4.1158	**4.1712**
	$Accuracy$	0.7255	**0.7286**	0.8857	**0.8861**
	R^2	0.8436	**0.8474**	0.7679	**0.7694**
	var	0.8436	**0.8474**	0.7684	**0.7705**
60min	$RMSE$	4.0099	**3.9707**	7.2677	**7.099**
	MAE	2.7860	**2.7391**	4.6021	**4.2343**
	$Accuracy$	0.7242	**0.7269**	0.8762	**0.8790**
	R^2	0.8421	**0.8454**	0.7283	**0.7407**
	var	0.8421	**0.8454**	0.7290	**0.7415**

3.4 Perturbation analysis

Noise is inevitable in real-world datasets. Therefore, perturbation analysis is conducted to test the robustness of A3T-GCN. In this experiment, two types of random noises are added to the traffic data. Random noise obeys Gaussian distribution $N \in (0, \sigma^2)$, where $\sigma \in (0.2, 0.4, 0.8, 1, 2)$, and Poisson distribution $P(\lambda)$ where $\lambda \in (1, 2, 4, 8, 16)$. The noise matrix values are normalized to [0,1].

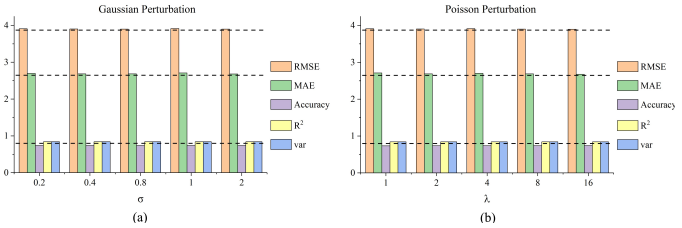


Fig. 6. SZ-taxi: perturbation analysis.

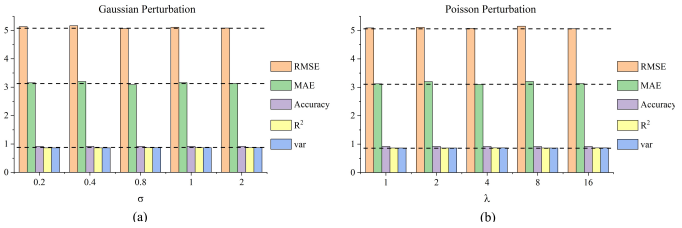
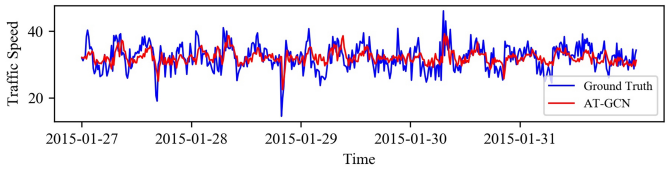
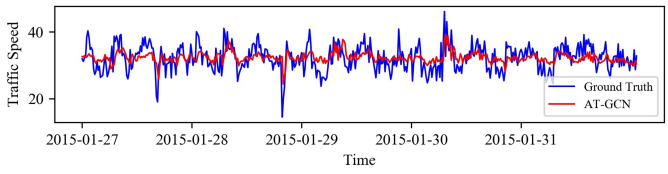


Fig. 7. Los-loop: perturbation analysis.

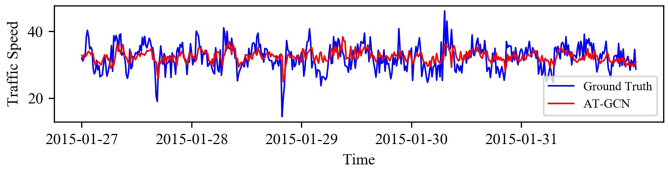
The experimental results based on SZ_taxi are shown in Fig. **??**. The results of adding Gaussian noise are shown in



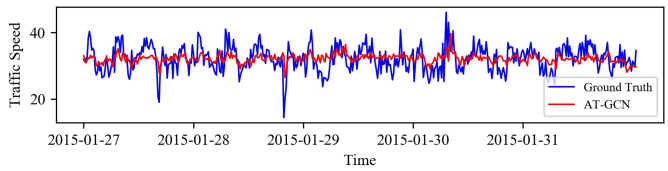
(a) 15 minutes



(b) 30 minutes

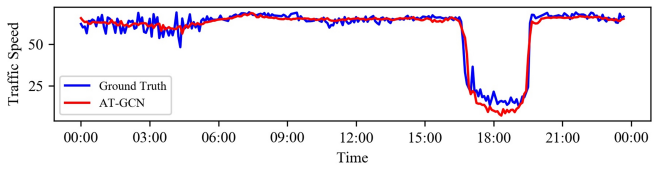


(c) 45 minutes

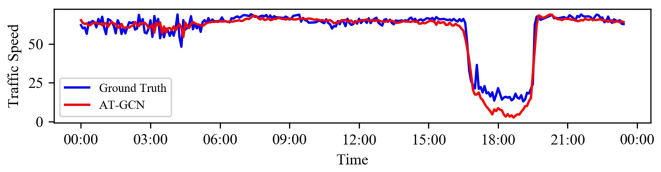


(d) 60 minutes

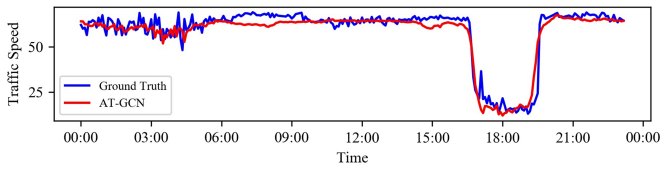
Fig. 8. The visualization results for prediction horizon of 15, 30, 45, 60 minutes (SZ-taxi).



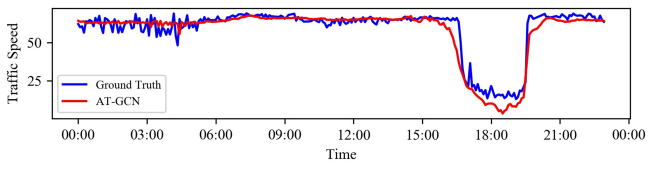
(a) 15 minutes



(b) 30 minutes



(c) 45 minutes



(d) 60 minutes

Fig. 9. The visualization results for prediction horizon of 15, 30, 45, 60 minutes (Los-loop).

Fig. **??**(a), where the x- and y-axes show the changes in σ and in different evaluation metrics, respectively. Different colors represent various metrics. Similarly, the results of adding Poisson noise are shown in Fig. **??**(b). The values of different evaluation metrics remain basically the same regardless of the changes in σ/λ. Hence, the proposed model can remarkably resist noise and process strong noise problems.

The experimental results based on Los_loop are consistent with experimental results based on SZ_taxi (Fig. **??**). Therefore, the A3T-GCN model can remarkably resist noise and still obtain stable forecasting results under Gaussian and Poisson perturbations.

3.5 Visualized analysis

The forecasting results of A3T-GCN model based on two real datasets are visualized for a good explanation of the model.

(1) SZ-taxi: We visualize the result of one road on January 27, 2015. Visualization results in 15, 30, 45, and 60 minutes of time series are shown in Fig. **??**.

(2) Los-loop: Similarly, we visualize one loop detector data in Los-loop dataset. Visualization results in 15, 30, 45, and 60 minutes are shown in Fig. **??**.

In sum, the predicted traffic speed shows similar variation trend with actual traffic speed under different time series lengths, which suggest that the A3T-GCN model is competent in the traffic forecasting task. This model can also capture the variation trends of traffic speed and recognize the start and end points of rush hours. The A3T-GCN model forecasts traffic jam accurately, thereby proving its validity in real-time traffic forecasting.

4 Conclusions

A traffic forecasting method called A3T-GCN is proposed to capture global temporal dynamics and spatial correlations simultaneously and facilitates traffic forecasting. The urban road network is constructed into a graph, and the traffic speed on roads is described as attributes of nodes on the graph. In the proposed method, the spatial dependencies are captured by GCN based on the topological characteristics of the road network. Meanwhile, the dynamic vari-

ation of the sequential historical traffic speeds is captured by GRU. Moreover, the global temporal variation trend is captured and assembled by the attention mechanism. Finally, the proposed A3T-GCN model is tested in the urban road network-based traffic forecasting task using two real datasets, namely, SZ-taxi and Los-loop. The results show that the A3T-GCN model is superior to HA, ARIMA, SVR, GCN, GRU, and T-GCN in terms of prediction precision under different lengths of prediction horizon, thereby proving its validity in real-time traffic forecasting.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation of China [grant numbers 41571397, 41501442, 41871364, 51678077 and 41771492].

REFERENCES

[1] Mohamed S. Ahmed and Allen R. Cook. *ANALYSIS OF FREEWAY TRAFFIC TIME-SERIES DATA BY USING BOX-JENKINS TECHNIQUES*. 1979.

[2] N. S. Altman. An introduction to kernel and nearest-neighbor nonparametric regression. *American Statistician*, 46(3):175–185, 1992.

[3] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2014.

[4] . Bengio, Y., . Simard, P., and . Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans Neural Netw*, 5(2):157–166, 2002.

[5] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann Lecun. Spectral networks and locally connected networks on graphs. 2013.

[6] Xiaofeng Cao, Yuhua Zhong, Zhou Yun, Wang Jiang, and Weiming Zhang. Interactive temporal recurrent convolution network for traffic prediction in data centers. *IEEE Access*, PP(99):1–1, 2017.

[7] Kyunghyun Cho, Bart Van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *Computer Science*, 2014.

[8] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014.

[9] Michal Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering, 2016.

[10] Chun Jiao Dong, Chun Fu Shao, Zhuge Cheng-Xiang, and Meng Meng. Spatial and temporal characteristics for congested traffic on urban expressway. *Journal of Beijing University of Technology*, 38(8):1242–1246+1268, 2012.

[11] Gui Fu, GuoQiang Han, Feng Lu, and ZiXin Xu. Short-term traffic flow forecasting model based on support vector machine regression. *Journal of South China University of Technology*, 41(9):71–76, 2013.

[12] Lei Gao, Xingquan Liu, Yu Liu, Pu Wang, Min Deng, Qing Zhu, and Haifeng Li. Measuring road network topology vulnerability by ricci curvature. *Physica A: Statistical Mechanics and its Applications*, 527:121071, 2019.

[13] Alex Graves. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[14] Victoria J. Hodge, Rajesh Krishnan, Jim Austin, John Polak, and Tom Jackson. Short-term prediction of traffic flow using a binary neural network. *Neural Computing and Applications*, 25(7-8):1639–1655, 2014.

[15] Hai Jun Huang. Dynamic modeling of urban transportation networks and analysis of its travel behaviors. *Chinese Journal of Management*, 2005.

[16] Yuan Jian and Bingquan Fan. Synthesis of short-term traffic flow forecasting research progress. *Urban Transport of China*, 2012.

[17] Liu Jing and Guan Wei. A summary of traffic flow forecasting methods. *Journal of Highway and Transportation Research & Development*, 2004.

[18] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks, 2016.

[19] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Graph convolutional recurrent neural network: Data-driven traffic forecasting. *CoRR*, abs/1707.01926, 2017.

[20] J. W. C. Van Lint, S. P. Hooqendoorn, and H. J. Van Zuvlen. Freeway travel time prediction with state-space neural networks: Modeling state-space dynamics with recurrent neural networks. *Transportation Research Record Journal of the Transportation Research Board*, 1811(1):347–369, 2002.

[21] Minh Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. *Computer Science*, 2015.

[22] M Morav?ík, M Schmid, N Burch, V Lisy, D Morrill, N Bard, T Davis, K Waugh, M Johanson, and M Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337):eaam6960, 2017.

[23] Iwao Okutani and Yorgos J. Stephanedes. Dynamic prediction of traffic volume through kalman filtering theory. *Transportation Research Part B Methodological*, 18(1):1–11, 1984.

[24] Nikolaos Pappas and Andrei Popescu-Belis. Multilingual hierarchical attention networks for document classification. 2017.

[25] Fu Rui, Zhang Zuo, and Li Li. Using lstm and gru neural network methods for traffic flow prediction. In *Youth Academic Conference of Chinese Association of Automation*, 2016.

[26] D Silver, J Schrittwieser, K Simonyan, I Antonoglou, A. Huang, A Guez, T Hubert, L Baker, M. Lai, and A Bolton. Mastering the game of go without human knowledge. *Nature*, 550(7676):354–359, 2017.

[27] Hongyu Sun, Chunming Zhang, and Bin Ran. Interval prediction for traffic time series using local linear predictor. In *International IEEE Conference on Intelligent Transportation Systems*, 2004.

[28] Shiliang Sun, Changshui Zhang, and Guoqiang Yu. A bayesian network approach to traffic flow forecasting. *IEEE Transactions on Intelligent Transportation Systems*, 7(1):124–132, 2006.

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[30] Chun-Hsin Wu, Jan-Ming Ho, and D. Lee. Travel-time prediction with support vector regression. *Intelligent Transportation Systems, IEEE Transactions on*, 5:276 – 281, 01 2005.

[31] Yuankai Wu and Huachun Tan. Short-term traffic flow forecasting with spatial-temporal correlation in a hybrid deep learning framework. 2016.

[32] Jun Xiao, Hao Ye, Xiangnan He, Hanwang Zhang, Fei Wu, and Tat-Seng Chua. Attentional factorization machines: Learning the weight of feature interactions via attention networks, 2017.

[33] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention, 2015.

[34] Hongbin Yin, S. C. Wong, Jianmin Xu, and C. K. Wong. Urban traffic flow prediction using a fuzzy-neural approach. *Transportation Research Part C*, 10(2):85–98, 2002.

[35] Byeonghyeop Yu, Yongjin Lee, and Keemin Sohn. Forecasting road traffic speeds by considering area-wide

spatio-temporal dependencies based on a graph convolutional neural network (gcn). *Transportation Research Part C: Emerging Technologies*, 114:189–204, 2020.

[36] H. Yu, Z. Wu, S. Wang, Y. Wang, and X. Ma. Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks. *Sensors*, 17(7):1501–, 2017.

[37] Ling Zhao, Yujiao Song, Chao Zhang, Yu Liu, Pu Wang, Tao Lin, Min Deng, and Haifeng Li. T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, 2019.