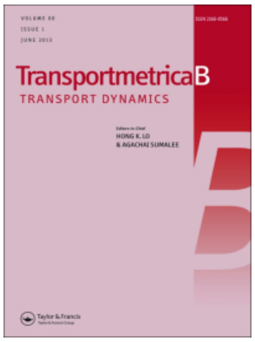


Transportmetrica B: Transport Dynamics

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/ttrb20

Graph attention temporal convolutional network for traffic speed forecasting on road networks

Ke Zhang , Fang He , Zhengchao Zhang , Xi Lin & Meng Li

To cite this article: Ke Zhang , Fang He , Zhengchao Zhang , Xi Lin & Meng Li (2021) Graph attention temporal convolutional network for traffic speed forecasting on road networks, Transportmetrica B: Transport Dynamics, 9:1, 153-171, DOI: 10.1080/21680566.2020.1822765

To link to this article: https://doi.org/10.1080/21680566.2020.1822765

Published online: 23 Sep 2020.

TRANSPORTMETRICA B: TRANSPORT DYNAMICS
2021, VOL. 9, NO. 1, 153–171
https://doi.org/10.1080/21680566.2020.1822765

Taylor & Francis
Taylor & Francis Group

Graph attention temporal convolutional network for traffic speed forecasting on road networks

Ke Zhang[a], Fang He[b,c], Zhengchao Zhang[a], Xi Lin[a,c] and Meng Li[a,c]

[a]Department of Civil Engineering, Tsinghua University, Beijing, People's Republic of China; [b]Department of Industrial Engineering, Tsinghua University, Beijing, People's Republic of China; [c]Tsinghua-Daimler Joint Research Center for Sustainable Transportation, Tsinghua University, Beijing, People's Republic of China

ABSTRACT

Traffic speed forecasting plays an increasingly essential role in successful intelligent transportation systems. However, this still remains a challenging task when the accuracy requirement is demanding. To improve the prediction accuracy and achieve a timely performance, the capture of the intrinsically spatio-temporal dependencies and the creation of a parallel model architecture are required. Accordingly, we propose a novel end-to-end deep learning framework named Graph Attention Temporal Convolutional Network (GATCN). The proposed model employs the graph attention network to mine the complex spatial correlations within the traffic network and temporal convolution operation to capture temporal dependencies. In addition, the multi-head self-attention mechanism is incorporated into the model to extract the spatio-temporal coupling effects. Experiments show that the proposed model consistently outperforms other state-of-the-art baselines for various prediction intervals on two real-world datasets. Moreover, we reveal that the proposed model can effectively distinguish the sophisticated traffic patterns of ramps on expressways by analyzing the graph attention heatmap.

ARTICLE HISTORY
Received 31 August 2019
Accepted 8 September 2020

KEYWORDS
Traffic prediction; deep learning; graph attention network; temporal convolutional network

1. Introduction

Traffic condition forecasting is an indispensable constituent of traffic modeling, operation, and management. High-precision traffic prediction information plays an important role in numerous intelligent transportation system (ITS) applications, including online traffic state information systems (Papathanasopoulou, Markou, and Antoniou 2016), travelers' route planning and departure time scheduling (Yuan et al. 2011), dynamic traffic signal optimization (Xu et al. 2018; Hao et al. 2019; Ahmed et al. 2019), and ramp control (Goatin, Göttlich, and Kolb 2016). To build a reliable prediction, it is not only necessary to extract the temporal information from inherent observation times-series, but it is also important to incorporate spatial information in the road network. During the past decade, significant attention has been attracted to model the temporal and spatial evolution of traffic circulation under the topic of prediction, and we summarize these in the following section.

The techniques for extracting temporal information can be mainly divided into two categories: classical statistical models and machine learning methods. Most statistical methods are based on

CONTACT Meng Li mengli@tsinghua.edu.cn Department of Civil Engineering, Tsinghua University, Beijing 100084, People's Republic of China; Tsinghua-Daimler Joint Research Center for Sustainable Transportation, Tsinghua University, Beijing 100084, People's Republic of China

some assumptions and have fixed components such as the autoregressive integrated moving average (ARIMA) and its many variants (Lippi, Bertini, and Frasconi 2013; Wang et al. 2016). The parameters of the models can be calibrated with empirical data through parametric estimation (e.g. maximum likelihood). However, due to the stochastic and nonlinear characteristics of traffic flow, it is difficult to overcome the limitations of such statistical models.

With the advent of the big data era, the popular introduction of advanced data management systems has made tremendous quantities of traffic data available. This wealth of traffic data, in turn, has rendered promising opportunities to data-driven prediction approaches. The support vector regression model, K-nearest neighbor model and gradient boosting regression tree tend to outperform the statistical methods (Wu, Ho, and Lee 2004; Cai et al. 2016; Zhan et al. 2019). Furthermore, deep learning methods have reported better performance than the aforementioned models. Deep belief networks (DBN) optimized by the multi-objective particle swarm algorithm can boost the accuracy of the forecasting result (Li et al. 2019). The recurrent neural network (RNN) and its variants (e.g. long short-term memory (LSTM) and gated recurrent unit (GRU)) are good at depicting temporal variables step by step through hidden states (Ma et al. 2015; Vinayakumar, Soman, and Poornachandran 2017). Bayesian combination model with deep learning has been proven to learn high-dimensional features effectively, and demonstrated excellent potentials (Gu et al. 2019b). However, these models are primarily aimed at learning temporal relevance from a single sequence, and neglect the spatial information in traffic networks.

Since the traffic statuses at downstream roads and upstream roads affect each other through transfer and feedback effects (Dong et al. 2012), it is necessary to entail the spatial correlations into the prediction paradigm. However, a traffic network is essentially based on a topology graph that is typically non-Euclidean structured data (Chang et al. 2019). Thus, directly utilizing correlation coefficient-based tools (Gu et al. 2019a) or convolutional neural networks (CNNs, Ma et al. 2017) will impair the capacity of the spatial dependencies analysis in traffic prediction models. To fill this gap, graph convolution networks (GCNs) are proposed to deal with a general data class defined on arbitrary topology structures (Kipf and Welling 2016).

Meanwhile, GCNs have attracted substantial attention in the transportation field. A spectral-based graph convolution (Defferrard, Bresson, and Vandergheynst 2016) was adopted and combined with the RNN (Li et al. 2017) and CNN (Yu, Yin, and Zhu 2017) to forecast traffic states. Non-spectral approaches have also been innovatively deployed in traffic speed forecasting. The traffic graph convolutional recurrent neural network (TGC-LSTM) is proposed to learn high-order neighborhood relationships in the traffic network that also considers travel time to measure the adjacency (Cui et al. 2019). Furthermore, graph convolutional network and temporal attention mechanism are integrated into a Seq2Seq framework to depict the spatio-temporal correlations in multistep traffic prediction (Zhang et al. 2019).

To summarize, previous studies have gained fruitful results in the domain of spatio-temporal dependencies modeling. However, with the rapid development of deep learning, emerging technologies can be adopted to develop high precision and computationally efficient prediction architectures. In the field of sequence learning, previous research has commonly regarded the RNN as the default starting point for sequential data modeling. More recent works have tended to consider utilizing temporal convolutional network (TCN) for time-series problems, which is a more lightweight framework (Bai, Kolter, and Koltun 2018). TCN has exhibited superior performance to RNN across a diverse range of tasks and datasets, because of enabling longer effective sequential memory. As for the operation on graph-structured data, graph attention network (GAT) leverage masked self-attentional layers to specify different weights to different neighborhoods on a directed graph without any kind of costly matrix operation (Veličković et al. 2017). However, few studies have attempted to fuse these kinds of neural networks into a customized spatio-temporal learning approach.

In this work, we propose a novel deep learning structure named Graph Attention Temporal Convolutional Network (GATCN). The spatial and temporal dependencies are captured from the GAT and TCN, respectively; in addition, the multi-head self-attention mechanism is incorporated into the model to extract the spatio-temporal coupling effects and to refine the scalability. Multi-head self-attention allows the model to attend to information from different representation subspaces jointly at different positions, which may help to capture the temporal information of traffic flows (Vaswani et al. 2017). We compare our algorithm with several representative baselines on a real-world dataset in Beijing. The results show that our model outperforms them under different prediction intervals. Furthermore, the effect of the multi-head self-attention mechanism and graph attention network are validated through a sensitivity analysis. Finally, we demonstrate that the GATCN can identify the most influential areas in a road network structure, which corresponds to the ramps on expressways.

The rest of the paper is organized as follows. Section 2 describes some preliminaries, the overall architecture and mathematical formulation of the proposed GATCN. Section 3 introduces the numerical experiment and comparison results based on real data. Section 4 presents the detailed model analysis. Finally, we conclude the paper and outline the future work in Section 5.

2. Methodology

2.1. Preliminaries

In this section, we introduce notations for the variables in this paper, and briefly describe the traffic prediction problem.

The underlying road network can be represented by a directed graph $\mathcal{G}(\mathcal{N},\mathcal{E})$, where the vertex set $\mathcal{N}$ represents the split points (e.g. intersections and detectors) and the edge set $\mathcal{E}$ represents the road segments. $\boldsymbol{A} \in \mathbb{R}^{|\mathcal{E}|\times|\mathcal{E}|}$($|\mathcal{E}|$ is the cardinality of link set $\mathcal{E}$) is the adjacency matrix of road segments, in detail, $\boldsymbol{A_{i,j}} = 1$ if road segment e_j is adjacent to road segment e_i along the driving direction, else $\boldsymbol{A_{i,j}} = 0$. The K-hop neighborhood for road segment i is defined as $NB_i(K) = \{e_j \in \mathcal{E}\ |d(e_i, e_j) \leq K\}$, where $d(e_i, e_j)$ denotes the minimum number of needed split points among all the walks from e_i to e_j. The one-hop neighborhood matrix for graph $\mathcal{G}(\mathcal{N},\mathcal{E})$ is exactly the adjacency matrix $\boldsymbol{A}$.

The speed v_t^i at the tth time slot (e.g. 5 min) of road segment e_i is defined as the average speed of floating cars during this time interval on the road segment. The spatio-temporal variables matrix $[\boldsymbol{V}_1, \ldots, \boldsymbol{V_{t-1}}, \boldsymbol{V_t}]$ denotes the speed sequences distributed over the traffic network, where $\boldsymbol{V_t} = (v_t^1, v_t^2, \ldots, v_t^{|\mathcal{E}|})$. In this sense, traffic flow prediction is a multivariate time series problem to estimate the traffic speed at a future time (i.e. $[\hat{\boldsymbol{V}}_{t+1}, \ldots, \hat{\boldsymbol{V}}_{t+n}]$) based on the observations collected over previous periods from the overall road network (i.e. $[\boldsymbol{V_{t-m+1}}, \ldots, \boldsymbol{V_{t-1}}, \boldsymbol{V_t}]$, m refers to the look-back time window).

Some exogenous variables containing historical information, time-of-day, and weekend-or-weekday are introduced to help the prediction. Terms $v_{t,\mathrm{ave}}^i$, $v_{t,\mathrm{median}}^i$, $v_{t,\max}^i$, $v_{t,\min}^i$, d_t^i are defined as the historical average value, median value, maximum value, minimum value, and standard deviation at the tth time slot of road segment e_i, respectively. Term p_t encodes the 5-min time slot into an integer number, and q_t is a dummy variable to identify a weekday or weekend.

Given the road network topology structure $\mathcal{G}(\mathcal{N},\mathcal{E})$ and relevant previously observed data until time interval t, the nth-step traffic speed prediction problem aims to predict the traffic speed $\hat{\boldsymbol{V}}_{\boldsymbol{t+n}}$at time interval $t+n$:

$$\hat{\boldsymbol{V}}_{\boldsymbol{t+n}} = f([\boldsymbol{V_{t-m+1}}, \ldots, \boldsymbol{V_{t-1}}, \boldsymbol{V_t}]; \mathcal{G}(\mathcal{N},\mathcal{E})) \tag{1}$$

2.2. GATCN model

In this subsection, we propose the novel GATCN model that integrates the GAT, TCN, and multi-head self-attention into an end-to-end deep learning architecture for traffic speed prediction. The framework of the proposed GATCN model is shown in Figure 1. The GAT is adopted to aggregate the spatial variables in the underlying traffic road network, then the spatially fused features are fed to the TCN module which accumulates the temporal information through the flexible reception field. In the end,

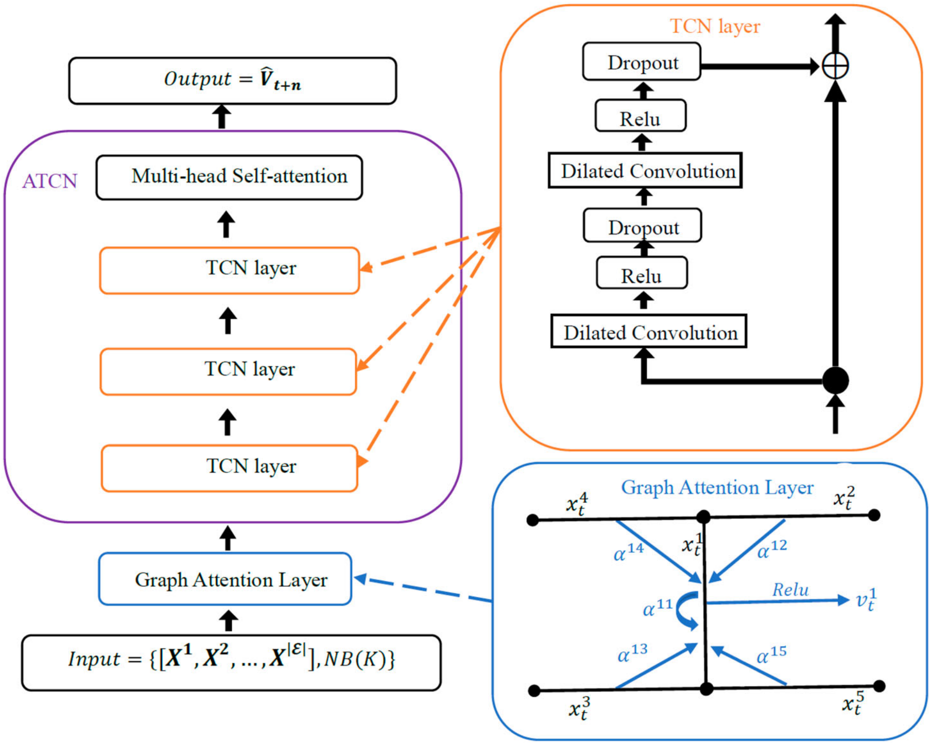


Figure 1. The architecture of the proposed GATCN model.

multi-head self-attention augments the spatio-temporal features by mixing them together through cross correlation coefficients. The specific details of the GATCN model are listed as follows:

2.2.1. Graph attention network

The GCN extends the applicable scope of standard convolution from a Euclidean space to a more general domain, which can be applied to learn the spatial dependencies in the road networks (Kipf and Welling 2016). As opposed to spectral-based GCN, GAT allows for assigning different importance to neighborhoods within the same hop in the style of the attention mechanism. The attention mechanism is applied in a shared manner to all links in the road network, and therefore it does not depend on upfront access to the global graph structure (Veličković et al. 2017). As shown in Figure 1, the graph attention layer is fed with a variety of explanatory variables $[\boldsymbol{X}^1, \boldsymbol{X}^2, \ldots, \boldsymbol{X}^{|\mathcal{E}|}]$ including the real-time records, time-of-day, weekday-or-weekend, and historical information:

$$\boldsymbol{X^i} = (v^i_{t,\text{ave}}, v^i_{t,\text{median}}, v^i_{t,\max}, v^i_{t,\min}, d^i_{\text{t}} || p_t, q_t || v^i_{t-m+1}, \ldots, v^i_{t-1}, v^i_t) \tag{2}$$

where $(\cdot||\cdot)$ denotes the concatenation of vectors or matrices across the column and $\boldsymbol{X^i} \in R^{m+7}$. To obtain a sufficiently expressive power, a shared linear transformation with learnable weight matrix $\boldsymbol{W}_1 \in \mathbb{R}^{F_1 \times |\mathcal{E}|}$ transforms the original input features into a high-dimensional representation space (F_1 is the cardinality of the embedding space). Then, Equation (3) computes the pair-wise attention score that indicates the importance of road segment j to i:

$$u^{ij}(K) = \text{LeakyRelu}(a(\boldsymbol{W}_1\boldsymbol{X^i}||\boldsymbol{W}_1\boldsymbol{X^j})), \quad j \in NB_i(K) \tag{3}$$

$$\text{LeakyRelu}(a) = \begin{cases} a \text{ if } a \geq 0 \\ \dfrac{a}{0.2} \text{ if } a < 0 \end{cases} \tag{4}$$

where $a(\cdot)$ conducts a linear mapping from matrix to scalar i.e. $\mathbb{R}^{2F_1\times(m+7)} \to \mathbb{R}$, LeakyRelu is a widely used activation function.

Once obtained, the attention scores are normalized by the softmax function which also makes coefficients easily comparable across different road segments:

$$\alpha^{ij}(K) = \text{softmax}\ (u^{ij}(K)) = \frac{exp(u^{ij}(K))}{\sum_{p\in NB_i(K)} \exp(u^{ip}(K))} \tag{5}$$

Finally, the normalized results are utilized to compute a weighted combination of the features corresponding to the K-hop neighborhoods, to serve as the final output features for each road segment of the GAT layer:

$$\boldsymbol{V^i}(K) = \text{Relu}\left(\sum_{j\in NB_i(K)} \alpha^{ij}(K)\boldsymbol{W}_2\boldsymbol{X^j}\right) \tag{6}$$

where Relu is an activation function as given by $\text{Relu}(x) = \max(x, 0)$, and $\boldsymbol{V^i}(K)$ is the spatial-fused feature vector for road segment e_j with the dimension depending on parameter matrix $W_2 \in \mathbb{R}^{F_2\times|\mathcal{E}|}$.

2.2.2. Temporal convolutional network

Unlike an array of related studies, we do not follow the routine of combining RNN structures. In contrast, the TCN is applied to explore the temporal regularities which is composed of a combination of dilations and residual connections with the causal convolutions. More importantly, the TCN has a backpropagation path different from the temporal direction of the sequence, thus overcoming the problems of a vanishing gradient and an inability for parallel computation, which are the major drawbacks of the RNN.

Two layers of dilated convolution function and residual connection make up a basic TCN layer, as shown in Figure 1. This model offers more flexibility and better control of memory size through changing its receptive field size, principally by increasing the filter size, stacking more residual blocks, or using larger dilation factors. The key components in dilated convolution and residual connection are introduced as follows:

Due to the long dependency on sequence tasks, the TCN employs a dilated convolution that enables an exponentially large receptive field. More formally, for sequence $\mathbf{V}(K) = [\boldsymbol{V_{t-m}}_{-6}(K), \ldots, \boldsymbol{V_t}_{-1}(K), \boldsymbol{V_t}(K)]$ and convolution kernel $d : \{0, \ldots, \gamma - 1\}$, the dilated convolution operation F is defined as

$$F(\boldsymbol{V_j}(K)) = \sum_{i=0}^{\gamma-1} d(i) * \boldsymbol{V_{j-r*i}}(K) \tag{7}$$

where γ is kernel size, and $j - r * i$ is the location of the element where the dilated convolution operation is performed. The dilations are typically set to double every residual block, so the effective receptive size grows exponentially with the network depth. This helps to capture an increasing amount of the global context without increasing the size of the parameters. As shown in Figure 2, dilated convolutions with different dilation factors and kernel sizes are applied. For example, while r becomes 2 in Figure 2(a), one input is skipped in each step, and the dilation is equivalent to introducing a fixed step between every two adjacent filter taps.

The residual connection proposed by He et al. (2016) is used to speed up convergence, and enable the training of deeper models. As shown in Figure 3, the residual connection contains an identity map

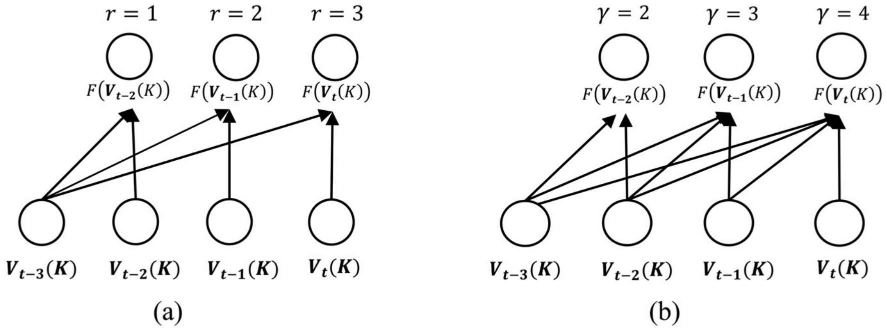


Figure 2. Dilated convolutions with different dilation factor r and kernel size γ. (a) Different r with constant $\gamma = 2$, (b) Different γ with constant $r = 1$.

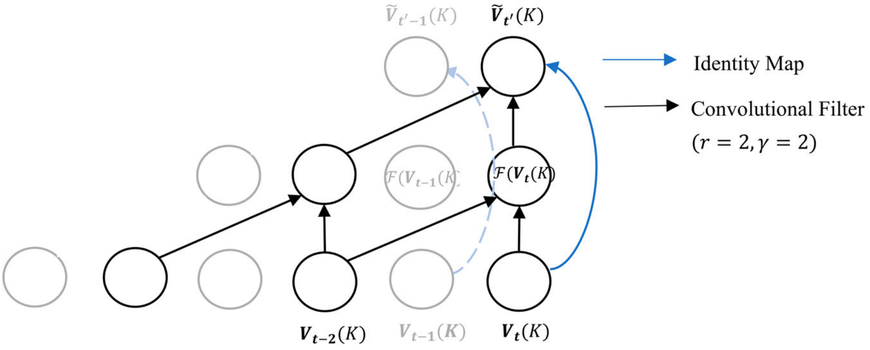


Figure 3. TCN layer.

leading out to a series of transformations $\mathcal{F}$, whose outputs are added to the input $\boldsymbol{V}(K)$ of the block:

$$\tilde{\boldsymbol{V}}(K) = F(\mathcal{F}(\boldsymbol{V}(K)) + V(K)) \tag{8}$$

$$F(\boldsymbol{V}(K)) = \text{Relu}(F(\boldsymbol{V}(K))) \tag{9}$$

where $\mathcal{F}(\cdot)$ represents the dilated convolution with non-linearity. In addition, a weight normalization (Salimans and Kingma 2016) is applied to the convolutional filters for normalization, and a spatial dropout is added after each dilated convolution for regularization.

We stack three TCN layers in our model; $\tilde{\boldsymbol{V}}(K)$ is the output of first TCN layer and also the input of the second TCN layer. For the sake of brevity, we denote the output of the third TCN layer as $\tilde{\boldsymbol{V}}^{(3)}(K)$ directly.

2.2.3. Multi-head self-attention

To acquire the coupling of spatio-temporal dependencies, multi-head self-attention boosts the representation of the output features after the TCN layer. In detail, dot-product attention function can be described as mapping a query and a set of key-value pairs to an output, in which the query, keys, values, and output are all vectors or matrices (Luong, Pham, and Manning 2015). The output is computed as a weighted sum of the values, in which the weight assigned to each value is computed by a compatibility function of the query with the corresponding key. In this case, the query, key and value all take on $\tilde{\boldsymbol{V}}^{(3)}(K)$ which is essentially a self-attention to gain the cross-correlation, as given by the

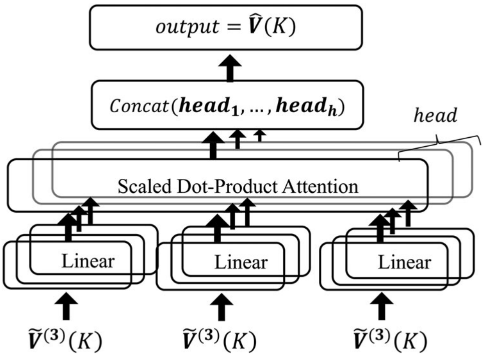


Figure 4. The multi-head self-attention layer.

following:

$$G(\tilde{\boldsymbol{V}}^{(3)}(K)) = \text{softmax}\left(\frac{\tilde{\boldsymbol{V}}^{(3)}(K)\tilde{\boldsymbol{V}}^{(3)}(K)^T}{\sqrt{d_{\tilde{\boldsymbol{V}}}}}\right)\tilde{\boldsymbol{V}}^{(3)}(K) \tag{10}$$

where $d_{\tilde{\boldsymbol{V}}}$ is the vertical dimension of $\tilde{\boldsymbol{V}}^{(3)}(K)$, which is used to scale dot products and avoid an overflow of numerical calculations.

Instead of performing a single attention function, Vaswani et al. (2017) found it beneficial to project the queries, keys, and values linearly many times, with different learned linear projections. In simple terms, multi-head self-attention (as shown in Figure 4) allows the model to attend to information jointly from different representation subspaces at different positions:

$$\hat{\boldsymbol{V}}(K) = [\boldsymbol{head}_1(K), \ldots, \boldsymbol{head_h}(K)]\boldsymbol{W^o} \tag{11}$$

$$\boldsymbol{head_i}(K) = G(\tilde{\boldsymbol{V}}^{(3)}(K)\boldsymbol{W_i}^{\tilde{\boldsymbol{V}}^{(3)}(K)}) \tag{12}$$

where the operator $[\ \cdot\ , \cdot\ ]$ concatenates two tensors along the same dimensions, $\boldsymbol{W^o}$ and $\boldsymbol{W_i}^{\tilde{\boldsymbol{V}}^{(3)}(K)}$ are the parameter matrices of the projection.

2.2.4. Objective function

During the training phase, we use the Adam optimizer to train our model by minimizing the loss function between $\hat{\boldsymbol{V}}_{t+n}(K) = (\hat{\boldsymbol{v}}^1_{t+n}, \hat{\boldsymbol{v}}^2_{t+n}, \ldots, \hat{\boldsymbol{v}}^{|\mathcal{E}|}_{t+n})$ and V_{t+n} as given by

$$\text{loss}(\theta) = \frac{1}{|\mathcal{E}|}\sum_{i=1}^{|\mathcal{E}|}|\hat{\boldsymbol{v}}^i_{t+n} - \boldsymbol{v}^i_{t+n}| \tag{13}$$

where θ are all learnable parameters in the proposed model.

The training steps of the GATCN model is illustrated in Algorithm 1.

3. Experiments

3.1. Dataset description and preprocessing

The efficacy of the proposed ATCN model is examined through two real-world traffic datasets:

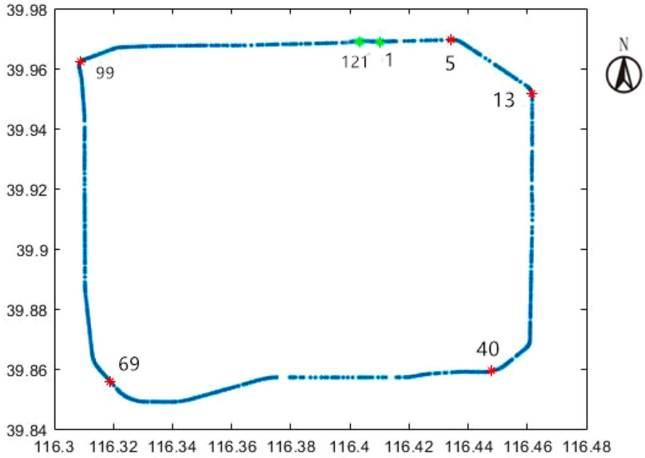


Figure 5. Road segments on 3rd ring road.

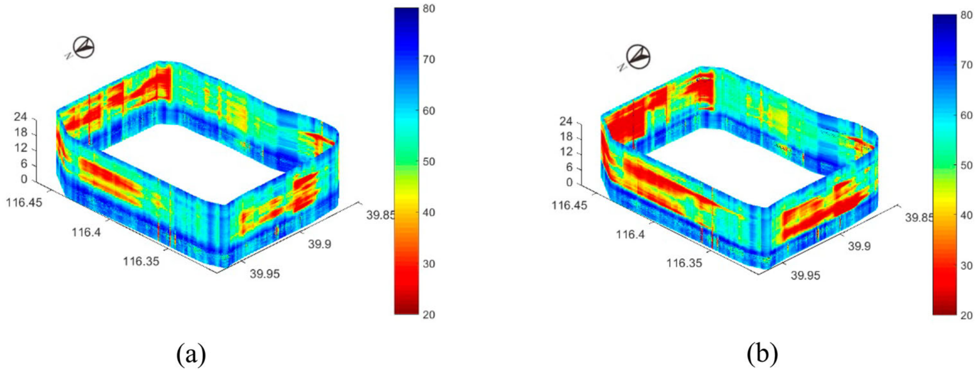


Figure 6. Traffic speed color map of 3rd ring road. (a) Weekends in July 2016, (b) Weekdays in July 2016.

3.1.1. 3rd ring dataset

This dataset is collected from the anonymous users of the smart-phone based AMAP APP (https://www.amap.com/) on the entire 3rd ring road in Beijing. As shown in Figure 5, the 48 km long 3rd ring road is partitioned into 121 road segments of length 400 m. Next, we calculate the 5 min average speed for each link by making use of the instantaneous state of navigation vehicles. The evolution plots of the traffic state in the 3rd ring road between weekdays and weekends are shown in Figure 6(a,b) with the *x*-axis for longitude, the *y*-axis for latitude, the *z*-axis for time and the colormap for speed. An apparent heterogeneity can be observed in the traffic status between weekdays and weekends. The whole dataset ranging from 1 October 2016 to 30 November 2016 is separated into three independent subsets, with the details presented in Table 1(a).

3.1.2. LA dataset

This traffic dataset contains traffic information collected from loop detectors in the highway of Los Angeles County (Jagadish et al. 2014). As shown in Figure 7, we select 146 sensors and collect 2 months of data ranging from 1 April 2012 to 30 May 2012 for the experiment. The details are presented in Table 1(b). Unlike the 3rd ring dataset, each traffic sensor in the LA dataset is considered as a vertex e_i and we compute the pairwise shortest distances between sensors to construct the topology graph.

Table 1. Descriptions of datasets.

	Training set	Validation set	Testing set
(a) 3rd ring dataset			
Date	Oct 1–Nov 20	Nov 21–Nov 23	Nov 24–Nov 30
Daily time range		06:00–22:00	
Size	$51\times192\times121\approx1.2$M	$3\times192\times121\approx70$K	$7\times192\times121\approx160$K
(b) LA dataset			
Date	Apr 1–May 20	May 21–May 23	May 24–May 30
Daily time range		06:00–22:00	
Size	$51\times192\times146\approx1.4$M	$3\times192\times146\approx84$K	$7\times192\times146\approx196$K

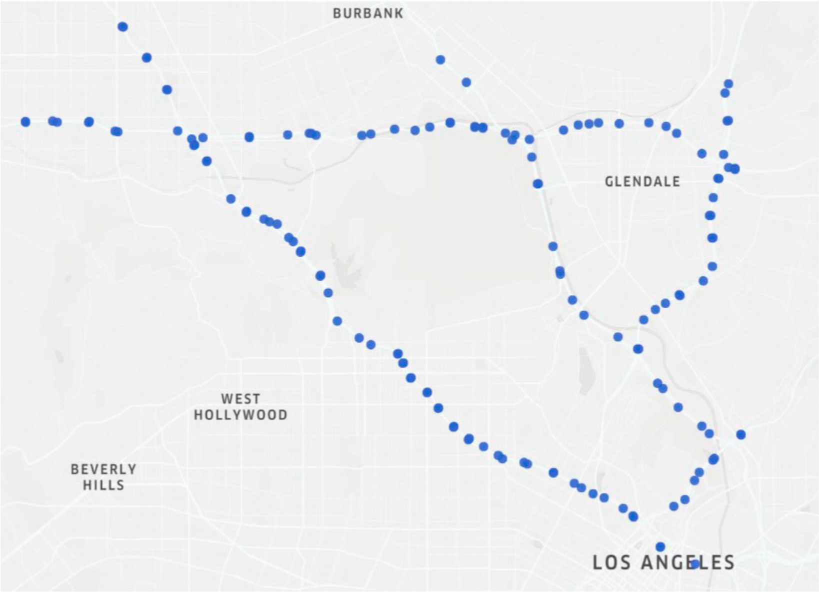


Figure 7. Traffic speed sensors on LA dataset.

The adjacency matrix is defined as

$$A = \begin{cases} \exp\left(-\frac{d^2_{e_i,e_j}}{\sigma^2}\right), & \text{if } \exp\left(-\frac{d^2_{e_i,e_j}}{\sigma^2}\right) \geq 0.1 \\ 0, \text{ otherwise} \end{cases} \tag{14}$$

where d_{e_i,e_j} is the shortest distance from sensor e_i to e_j, σ is the standard deviation of distances.

3.2. Experimental setting

In view of classical statistical methods always perform worse than machine learning methods on many traffic speed forecasting tasks, due to an inability to handle complex spatio-temporal information (Yu, Yin, and Zhu 2017; Zhang et al. 2019; Cui et al. 2019), we directly compare the GATCN with almost all prevailing neural network based methods. The baselines are boiled down to the following four types in view of the feature process:

Algorithm 1. GATCN model training

```
Input The road segment $i$, $K$-hop neighborhoods $NB(K)$, traffic speed of training dataset $[\boldsymbol{V_{t-m+1}}, \ldots, \boldsymbol{V_{t-1}}, \boldsymbol{V_t}]$, time-of-day of training dataset $\{p_{t-m+1}, \ldots, p_{t-1}, p_t\}$, weekday-or-weekend of training dataset $\{q_{t-m+1}, \ldots, q_{t-1}, q_t\}$, historical statistical information of training dataset $\{(v^i_{\tilde{t},\mathrm{ave}}, v^i_{\tilde{t},\mathrm{median}}, v^i_{\tilde{t},\mathrm{max}}, v^i_{\tilde{t},\mathrm{min}}, d^i_{\tilde{t}}), \quad \tilde{t} = t-m+1, \ldots, t-1, t\}$, prediction time step $t+n$
Output GATCN model with learnt parameters
Procedure GATCN model train
Define $\boldsymbol{X} = [\boldsymbol{X}^1, \boldsymbol{X}^2, \ldots, \boldsymbol{X}^{|\mathcal{E}|}]$, where $\boldsymbol{X^i} = (v^i_{t,\mathrm{ave}}, v^i_{t,\mathrm{median}}, v^i_{t,\mathrm{max}}, v^i_{t,\mathrm{min}}, d^i_{\mathrm{t}} || p_t, q_t || v^i_{t-m+1}, \cdots, v^i_{t-1}, v^i_t)$
Initialize all the weight and bias parameters
repeat
    Randomly extract a batch of samples from $\boldsymbol{X}$
    Compute spatial information $\boldsymbol{V}(\boldsymbol{K})$ by Equations (3–6) in GAT layer
    Compute temporal information $\tilde{\boldsymbol{V}}^{(3)}(K)$ by Equations (7–9) in TCN layers
    Compute outputs$\hat{\boldsymbol{V}}_{\boldsymbol{t+n}}$ by Equations (10–12) in multi-head self-attention layer
    Update the parameters by minimizing the objective function shown in Equation (13) through the Adam optimizer.
until convergence criterion met
end procedure
```

3.2.1. Classical baseline

Historical average values (HA, predicts the future speed based on the mean values of the same timestamp in the training set) and a fully-connected neural network (FCN) are served as the basic criteria among the extensive works (Rumelhart, Hinton, and Williams 1986).

3.2.2. Temporal model

The LSTM, GRU and TCN are three well-known weapons to address temporal correlations with the former two in the recurrent structure and the latter one in the style of convolution. The attention temporal convolutional network (ATGN) is evaluated to validate the effect of the multi-head self-attention mechanism. None of these take spatial features into account.

3.2.3. Spatial model

The spectral graph convolution (SGC) (Kipf and Welling 2016) and high-order adaptive graph convolutional network (HA-GCN) (Zhou and Li 2017) are two general-purposed architectures that conduct graph convolution to dynamically model spatial dependencies. The GAT provides a novel method to distinguish the spatial variables across the topology of the road network. A singe GAT layer does not carry out any calculations on the temporal dimension.

3.2.4. Spatio-temporal model

The traffic graph convolutional recurrent neural network (TGC-LSTM, Cui et al. 2019) conducts a convolution and LSTM to model the dynamics of the traffic flow which attains spatio-temporal dependencies.

The graph attention temporal convolutional network learns the complex spatio-temporal dependencies present in the traffic data, which is an improvement over the ATCN.

The speed records in the past 60 min (i.e. $m = 12$) are employed to forecast the traffic conditions in the next 5, 10, and 20 min. The missing rates for 3rd ring and LA dataset are 2.9% and 4.7%. To ensure more reliable results, the missing records were completed by the temporal nearest non-missing ones.

To guarantee the fairness of these experiments on neural network approaches, these models keep the same input feature vector and hyper-parameter settings (except for HA). They are all trained by Adam optimizer with batch size 64 and learning rate 10^{-3}. In the ATCN and GATCN, the number of attention heads is set as 5 and the filter size is set as 2. In the SGC, HA-GCN, GAT, TGC-LSTM and GATCN, the hop of neighborhoods is set as 1.

For choosing the hyperparameters of the TCN model, we perform a grid search to study the effect of different parameters: dropout probability (0, 0.2, 0.5), kernel size (2, 4), number of TCN layers (2, 3, 4). Finally, we choose the model that has the lowest RMSE on the validation dataset. In the TCN, ATCN, GATCN, the number of TCN layers is set as 3, the dropout probability is 0.5, kernel size is 2.

Our experiments are performed on a computing platform with characteristics as follows: NVIDIA Quadro P5000 with 16 GB memory, Intel(R) Xeon(R) CPU E5-2673 v3 @2.40 GHz with 256 GB RAM. Furthermore, we choose the PyTorch platform to implement the deep learning models. We have conducted experiments three times and reported the average results to avoid the impacts of randomness.

The performances of these models are assessed via three commonly used metrics in prediction problems, including the mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean squared error (RMSE), as given by

$$\mathrm{MAE} = \frac{1}{C}\sum_{i=1}^{C}|\hat{v}_t^i - v_t^i| \tag{15}$$

$$\mathrm{MAPE} = \frac{1}{C}\sum_{i=1}^{C}\left|\frac{v_t^i - \hat{v}_t^i}{v_t^i}\right| * 100\% \tag{16}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{C}\sum_{i=1}^{C}(\hat{v}_t^i - v_t^i)^2} \tag{17}$$

where v_t^i and $\hat{v}_t^i$ are the ground truth and prediction values at the ith road segment at time t, respectively, and C is the total number of observations.

3.3. Results

Tables 2 and 3 show the prediction errors of each model for 5-min, 10-min, and 20-min prediction intervals on the 3rd ring and LA dataset respectively. The following conclusions can be drawn from the results:

(1) Our proposed model achieves the best performance compared with other baselines in all three evaluation metrics during various steps.
(2) The HA cannot achieve good traffic prediction results as it relies on historical records to predict the future values without considering spatial, temporal and other related external features.
(3) In the comparison of temporal models, the TCN performs better than the generic recurrent architectures (i.e. LSTM, GRU) because it can set up a more effective temporal memory. The ATCN outperforms the first three models, which indicates that the multi-head self-attention can significantly promote the accuracy and also highlights the necessity of integrating it into our model.
(4) The GAT performs better than the SGC and HA-GCN, which emphasizes the effectiveness of the attention mechanism to capture spatial information for traffic speed forecasting.
(5) The TGC-LSTM is the most competitive benchmark. Although it brings spatio-temporal dependencies into consideration, the contrast proves that the GATCN is superior for traffic speed prediction according to three metrics. And GATCN costs less training time than TGC-LSTM, which mainly benefits from the parallel temporal convolution operation.
(6) All performances worsen as the prediction interval increases, because the temporal correlations of the traffic speeds between the time intervals being used to predict and the current moment decrease.

To analyze the prediction results of our model more intuitively, we visualize 5-min, 10-min, and 20-min samples of the predicted results on 3rd ring dataset in Figure 8. It can be easily found that they can all capture the general trend of the ground truth. However, in the peak hours, the LSTM and TGC-LSTM

Table 2. Prediction performances on 3rd ring dataset.

Model	MAE	RMSE	MAPE	Time(s)
(a) 5-min prediction horizon (one step)				
HA	8.04	11.09	31.91%	/
FCN	3.85	5.48	11.52%	59
LSTM	3.72	5.40	11.02%	148
GRU	3.74	5.40	10.93%	132
TCN	3.70	5.31	10.89%	118
ATCN	3.63	5.14	10.54%	162
HA-GCN	3.65	5.20	10.59%	180
SGC	3.68	5.28	10.72%	141
GAT	3.58	5.10	10.43%	202
TGC-LSTM	3.44	5.03	10.28%	412
GATCN	**3.32**	**4.78**	**9.89%**	342
(b) 10-min prediction horizon (two steps)				
HA	8.04	11.09	31.91%	/
FCN	4.81	7.25	14.94%	56
LSTM	4.66	7.03	14.68%	157
GRU	4.67	7.08	14.72%	142
TCN	4.65	6.97	14.55%	138
ATCN	4.53	6.72	14.27%	190
HA-GCN	4.58	6.73	14.42%	212
SGC	4.60	6.89	14.50%	192
GAT	4.48	6.65	14.12%	248
TGC-LSTM	4.42	6.62	13.89%	473
GATCN	**4.29**	**6.32**	**13.48%**	422
(c) 20-min prediction horizon (four steps)				
HA	8.04	11.09	31.91%	/
ANN	5.87	9.30	19.98%	57
LSTM	6.15	8.88	19.46%	221
GRU	6.03	8.79	19.30%	197
TCN	5.78	8.68	19.12%	179
ATCN	5.59	8.43	18.68%	234
HA-GCN	5.67	8.52	18.72%	221
SGC	5.72	8.66	18.92%	197
GAT	5.56	8.40	18.60%	251
TGC-LSTM	5.45	8.34	18.48%	563
GATCN	**5.38**	**8.04**	**17.96%**	471

lack the sensitivity with regard to the fluctuation of speed while the GATCN accurately depicts the sharp changes. This finding indicates that the GATCN can capture the variation characteristics of the ground truth more successfully.

4. Model analysis

To investigate the impact of the model structure and spatio-temporal information on the traffic speed prediction, we successively conduct analyses of the GAT, TCN module and multi-head self-attention in this section.

4.1. Interpretation of GAT

In this subsection, we explore the functions of the GAT on spatial information. First, the differences in K-hop neighborhoods on the GATCN are tested under the 10-min and 20-min prediction horizons on 3rd ring road network, as shown in Figure 9. In detail, the harvest is distinct by altering K from zero to one and relatively limited when K exceeds one. This is because a larger K enables the model to capture the broad spatial dependency between the predicted segment and its adjacent road segments. The

Table 3. Prediction performances on LA dataset.

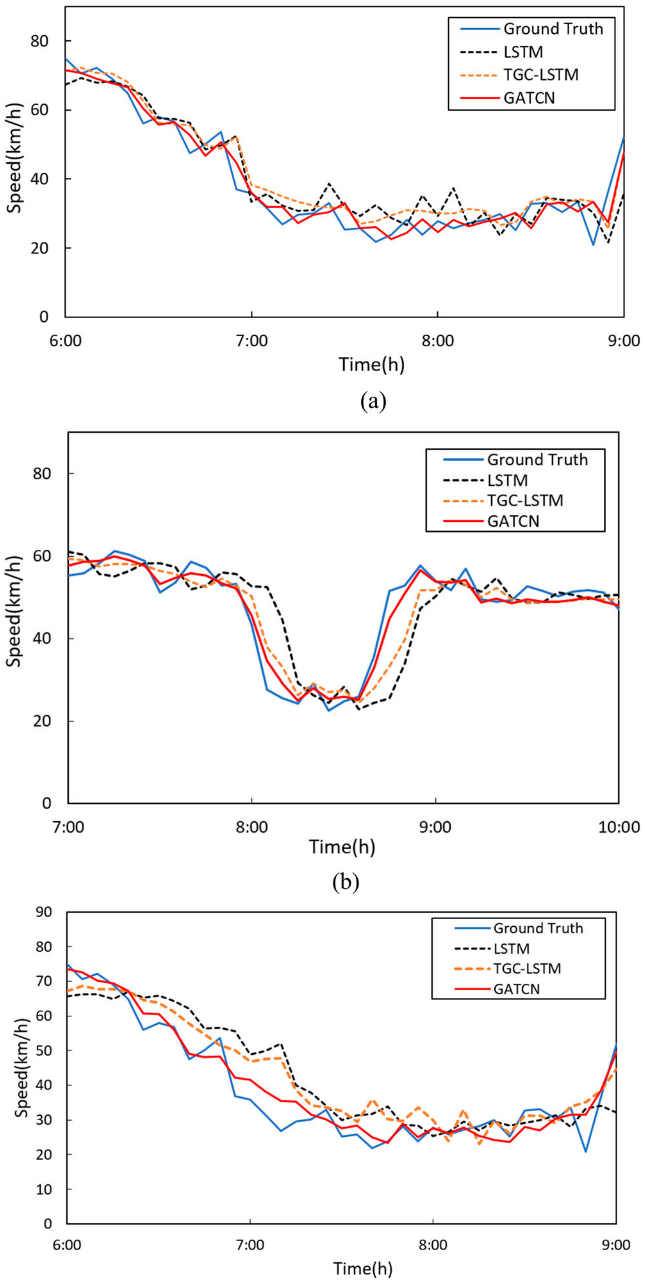
Model	MAE	RMSE	MAPE	Time(s)
(a) 5-min prediction horizon (one step)				
HA	7.38	11.55	22.43%	/
FCN	2.88	5.02	7.78%	66
LSTM	2.76	4.90	7.27%	160
GRU	2.80	4.95	7.32%	142
TCN	2.68	4.82	7.02%	131
ATCN	2.61	4.60	6.83%	186
HA-GCN	2.60	4.59	6.83%	177
SGC	2.68	4.68	6.95%	156
GAT	2.58	4.53	6.77%	223
TGC-LSTM	2.55	4.38	6.48%	433
GATCN	**2.39**	**4.17**	**6.14%**	372
(b) 10-min prediction horizon (two steps)				
HA	7.38	11.55	22.43%	/
FCN	3.34	6.55	9.28%	66
LSTM	3.13	6.09	8.87%	173
GRU	3.14	6.05	8.78%	167
TCN	3.11	5.94	8.56%	150
ATCN	3.05	5.79	8.43%	206
HA-GCN	3.09	5.62	8.57%	217
SGC	3.10	5.78	8.73%	186
GAT	3.01	5.47	8.48%	263
TGC-LSTM	2.95	5.37	8.29%	513
GATCN	**2.87**	**5.20**	**8.06%**	440
(c) 20-min prediction horizon (four steps)				
HA	7.38	11.55	22.43%	/
FCN	4.13	8.54	12.34%	60
LSTM	3.92	8.15	11.90%	213
GRU	3.95	8.23	12.03%	193
TCN	3.86	8.04	11.74%	184
ATCN	3.79	7.63	11.69%	255
HA-GCN	3.80	7.62	11.57%	227
SGC	3.81	7.78	11.73%	202
GAT	3.78	7.54	11.26%	263
TGC-LSTM	3.72	7.41	11.11%	603
GATCN	**3.68**	**7.20**	**10.95%**	514

result indicates that spatial information plays an important role in traffic prediction and characterizes the local correlations.

Moreover, to display the effect of the graph attention network more effectively, the heat maps of spatial attention weights (i.e. α^{ij}) in the GAT under different prediction horizons are presented in Figure 10. The salient regions in the heat map own higher coefficients, which indicates the spatial dependencies are more essential within the corresponding road segments. The spatial attention weights are smaller under 20-min prediction horizon because the spatial correlations decay with the increasing prediction horizon. Figure 11 traces the highlighted areas in the heat map to the sources of the physical locations on the 3rd ring road with different ellipses. We can find that the marked positions are mostly located at congested ramp entrances and overpasses. For example, the 'A' ellipse represents SanYuan bridge which contains several congested ramp entrances connecting to Xiangheyuan Road and the 'C' ellipse is on the overpass between 3rd Road and Nanyuan Road. It verifies that the GATCN picks up spatial features in an explicable way.

4.2. Parameter analysis of the TCN module

Figure 12 shows the hyper-parameter tuning of the TCN module in the GATCN on the 20-min traffic speed prediction in the validation set of 3rd ring road, in which four kinds of parameters are



(a)

(b)

(c)

Figure 8. Comparison of predictions with the ground truth on 3rd ring dataset. (a) 5-min traffic speed forecasting on Nov 24th (link 2), (b) 10-min traffic speed forecasting on Nov 26th (link 21), (c) 20-min traffic speed forecasting on Nov 24th (link 1).

investigated: the structure layers, dilation factor, kernel size, and dropout rate. The horizontal axis represents the parameter setting, and the vertical axis represents the change in the MAPE. From Figure 12(a), the MAPE first drops and then rises with the increase in TCN layers. This shows that a shallow structure (i.e. two layers) makes it difficult to capture complex temporal relationships, but with an

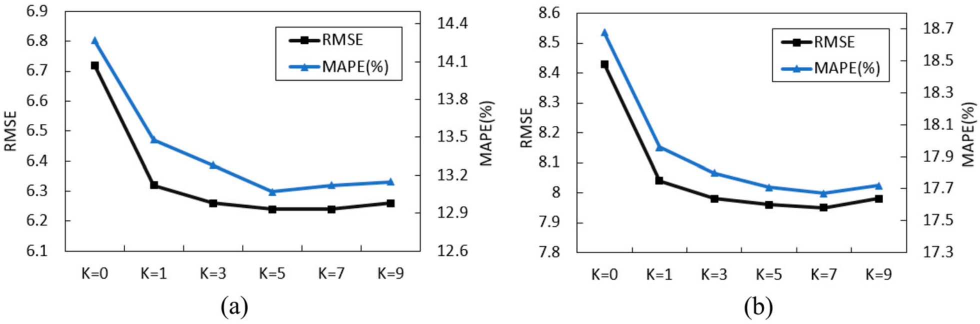


Figure 9. Influence of the *K*-hop neighborhoods on the GATCN. (a) 10-min traffic speed forecasting, (b) 20-min traffic speed forecasting.

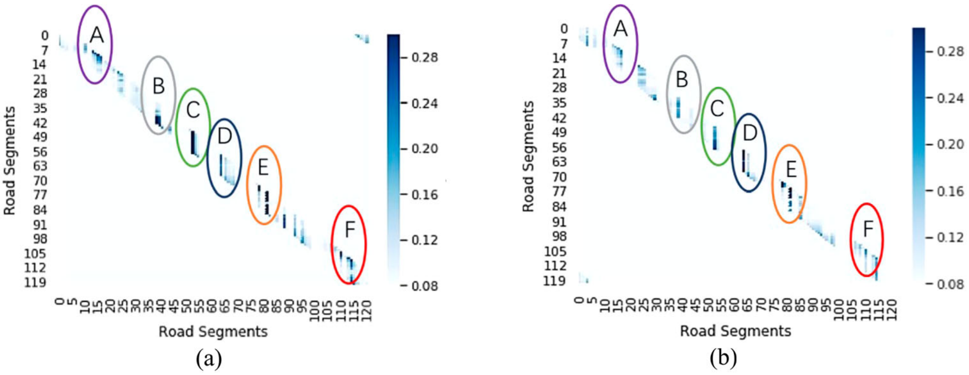


Figure 10. Heat map of the graph attention weight matrix. (a) 5-min prediction horizon, (b) 20-min prediction horizon.

increase in the depth, overfitting issues could occur. Figure 12(b) indicates that with large dilation factors, some useful information will be missed with the enlarged reception field, which leads to a worse performance. Figure 12(c,d) display the model transits from underfitting to overfitting with increases in kernel size and dropout rate, and the optimal parameter is the balance point between them.

4.3. Effect of multi-head self-attention

It was reported in Section 3.3 that the TCN without attention mechanism will sacrifice the predictive precision. In this subsection, we further evaluate the influences of the multi-head self-attention on the ATCN and GATCN in terms of the number of attention heads. The variation curves for the RMSE and MAPE with the attention heads in 10-min and 20-min traffic speed forecasting on the 3rd ring dataset are plotted in Figure 13. There is an obvious improvement when adding the attention head from zero to one. It is worth noting that the profit becomes inconspicuous with more attention heads because the computing time monotonously rises. This may be because the self-attention mechanism can effectively represent the probability of a relationship between the terms of a traffic speed sequence and find a new representation for each of the terms in the sequence for prediction. The contrast experiments demonstrate the effectiveness of applying multi-head self-attention to learn the spatio-temporal features in traffic predictions.

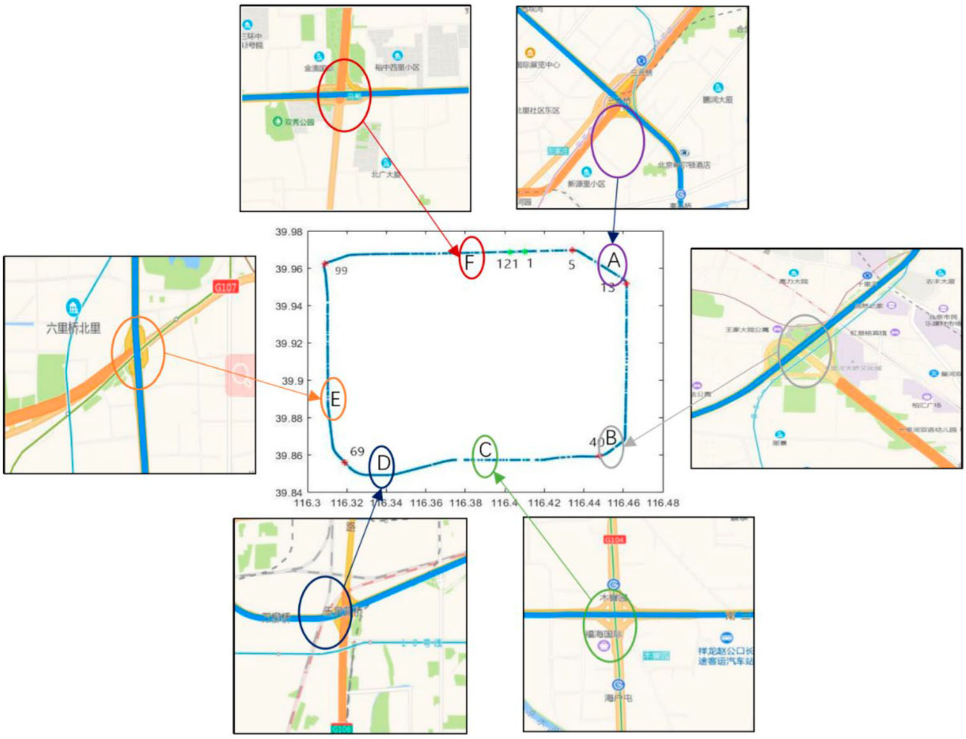


Figure 11. Marked regions on the heat map on 3rd ring road.

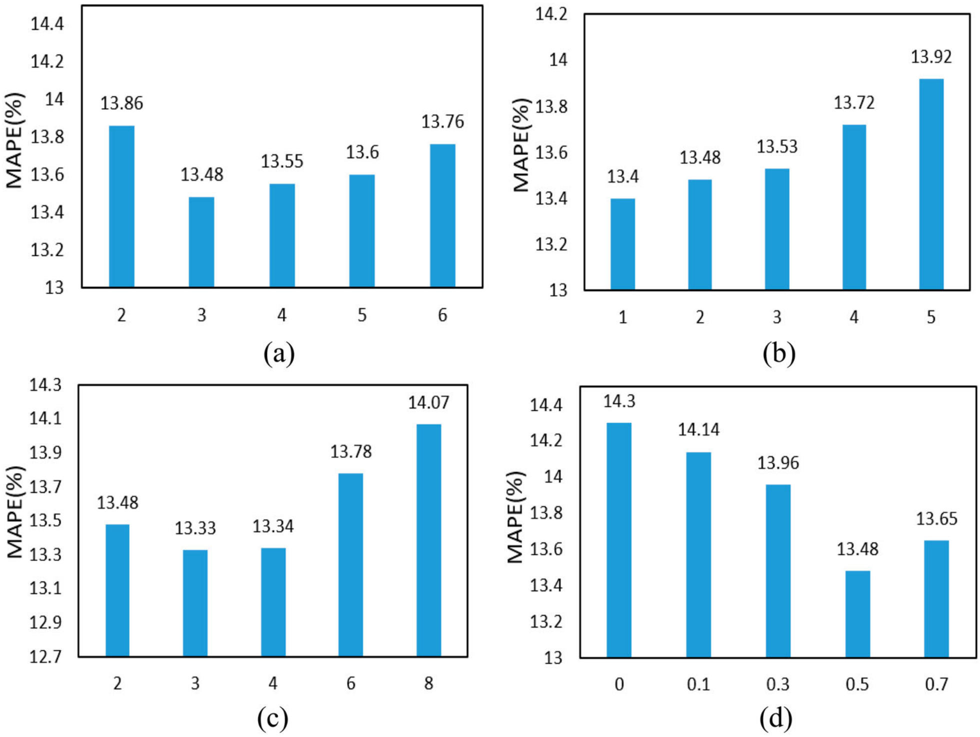


Figure 12. Hyper-parameter tuning on the TCN module. (a) The layers of TCN, (b) Dilation Factor, (c) Kernel size of TCN, (d) Dropout rate.

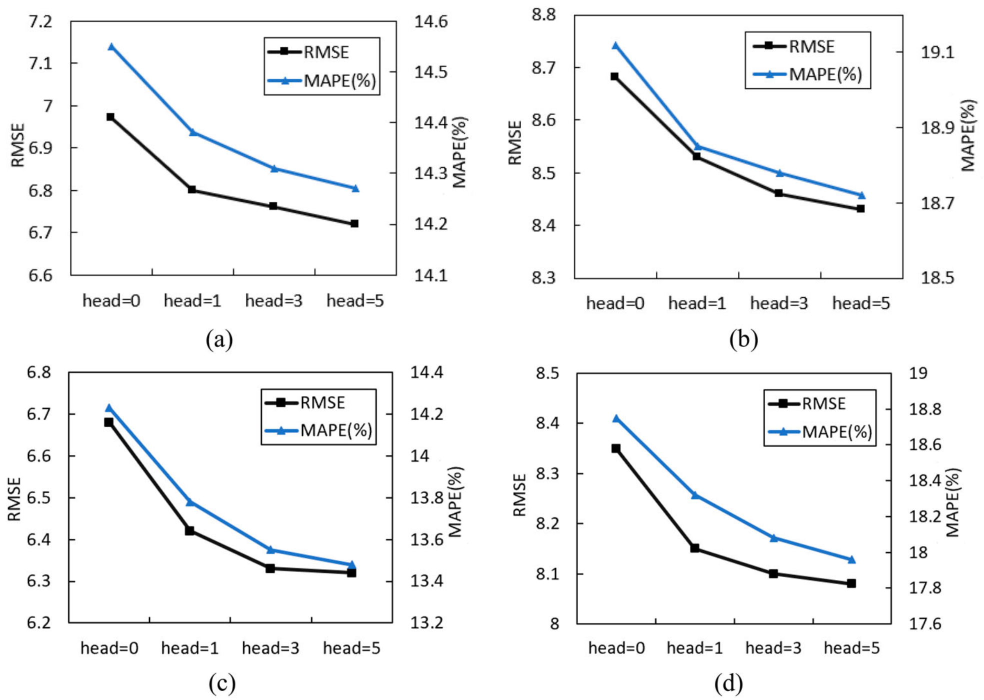


Figure 13. Influence of the self-attention heads. (a) 10-min prediction of the ATCN, (b) 20-min prediction of the ATCN, (c) 10-min prediction of the GATCN, (d) 20-min prediction of the GATCN.

5. Conclusions

In this paper, we propose a novel deep learning method (GATCN) to improve the performance for traffic speed forecasting on road networks. The GAT, TCN, and multi-head self-attention are subtly merged into a spatio-temporal learning framework. Through the evaluation on two real-world traffic datasets, we demonstrate the superiority of the proposed GATCN model compared with several well-known baselines. Furthermore, we discuss the influences of hyper-parameters including the K-hop neighborhoods, TCN layers, kernel size, dilation factor, dropout rate of the TCN and the number of attention heads on the GATCN through a sensitivity analysis. Finally, by comparing the attention weight matrix with the physical realities of the road network, it can be concluded that our model can reflect the complex spatial relevance in an interpretable way. For our future work, we will take advantage of the attention mechanism to incorporate the attributes of road segments, such as road class, number of lanes and traffic lights.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This research is supported by grants from National Key Research and Development Program of China [grant number 2018YFB1601600].

References

Ahmed, A., S. A. A. Naqvi, D. Watling, and D. Ngoduy. 2019. "Real-Time Dynamic Traffic Control Based on Traffic-State Estimation." *Transportation Research Record* 2673 (5): 584–595.

Bai, S., J. Z. Kolter, and V. Koltun. 2018. “An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling.” arXiv preprint arXiv:1803.01271.

Cai, P., Y. Wang, G. Lu, P. Chen, C. Ding, and J. Sun. 2016. “A Spatiotemporal Correlative k-Nearest Neighbor Model for Short-Term Traffic Multistep Forecasting.” *Transportation Research Part C: Emerging Technologies* 62: 21–34.

Chang, J., L. Wang, G. Meng, Q. Zhang, S. Xiang, and C. Pan. 2019. “Local-Aggregation Graph Networks.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Cui, Z., K. Henrickson, R. Ke, and Y. Wang. 2019. “Traffic Graph Convolutional Recurrent Neural Network: A Deep Learning Framework for Network-Scale Traffic Learning and Forecasting.” *IEEE Transactions on Intelligent Transportation Systems*.

Defferrard, M., X. Bresson, and P. Vandergheynst. 2016. “Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering.” *In Advances in Neural Information Processing Systems*, 3844–3852.

Dong, C. J., C. F. Shao, C. X. Zhuge, and M. Meng. 2012. “Spatial and Temporal Characteristics for Congested Traffic on Urban Expressway.” *Journal of Beijing Polytechnic University* 38 (8): 128–132.

Goatin, P., S. Göttlich, and O. Kolb. 2016. “Speed Limit and Ramp Meter Control for Traffic Flow Networks.” *Engineering Optimization* 48 (7): 1121–1144.

Gu, Y., W. Lu, L. Qin, M. Li, and Z. Shao. 2019a. “Short-Term Prediction of Lane-Level Traffic Speeds: A Fusion Deep Learning Model.” *Transportation Research Part C: Emerging Technologies* 106: 1–16.

Gu, Y., W. Lu, X. Xu, L. Qin, Z. Shao, and H. Zhang. 2019b. “An Improved Bayesian Combination Model for Short-Term Traffic Prediction with Deep Learning.” *IEEE Transactions on Intelligent Transportation Systems* 21: 1332–1342.

Hao, S., L. Yang, L. Ding, and Y. Guo. 2019. “Distributed Cooperative Backpressure-Based Traffic Light Control Method.” *Journal of Advanced Transportation*.

He, K., X. Zhang, S. Ren, and J. Sun. 2016. “Deep Residual Learning for Image Recognition.” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

Jagadish, H. V., J. Gehrke, A. Labrinidis, Y. Papakonstantinou, J. M. Patel, R. Ramakrishnan, and C. Shahabi. 2014. “Big Data and Its Technical Challenges.” *Communications of the ACM* 57 (7): 86–94.

Kipf, T. N., and M. Welling. 2016. “Semi-Supervised Classification with Graph Convolutional Networks.” arXiv preprint arXiv:1609.02907.

Li, L., L. Qin, X. Qu, J. Zhang, Y. Wang, and B. Ran. 2019. “Day-Ahead Traffic Flow Forecasting Based on a Deep Belief Network Optimized by the Multi-Objective Particle Swarm Algorithm.” *Knowledge-Based Systems* 172: 1–14.

Li, Y., R. Yu, C. Shahabi, and Y. Liu. 2017. “Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting.” arXiv preprint arXiv:1707.01926.

Lippi, M., M. Bertini, and P. Frasconi. 2013. “Short-term Traffic Flow Forecasting: An Experimental Comparison of Time-Series Analysis and Supervised Learning.” *IEEE Transactions on Intelligent Transportation Systems* 14 (2): 871–882.

Luong, M. T., H. Pham, and C. D. Manning. 2015. “Effective Approaches to Attention-Based Neural Machine Translation.” arXiv preprint arXiv:1508.04025.

Ma, X., Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang. 2017. “Learning Traffic as Images: A Deep Convolutional Neural Network for Large-Scale Transportation Network Speed Prediction.” *Sensors* 17 (4): 818.

Ma, X., Z. Tao, Y. Wang, H. Yu, and Y. Wang. 2015. “Long Short-Term Memory Neural Network for Traffic Speed Prediction Using Remote Microwave Sensor Data.” *Transportation Research Part C: Emerging Technologies* 54: 187–197.

Papathanasopoulou, V., I. Markou, and C. Antoniou. 2016. “Online Calibration for Microscopic Traffic Simulation and Dynamic Multi-Step Prediction of Traffic Speed.” *Transportation Research Part C: Emerging Technologies* 68: 144–159.

Rumelhart, D. E., G. E. Hinton, and R. J. Williams. 1986. “Learning Representations by Back-Propagating Errors.” *Nature* 323 (6088): 533–536.

Salimans, T., and D. P. Kingma. 2016. “Weight Normalization: A Simple Reparameterization to Accelerate Training of Deep Neural Networks.” *Advances in Neural Information Processing Systems*, 901–909.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. 2017. “Attention is All You Need.” *In Advances in Neural Information Processing Systems*, 5998–6008.

Veličković, P., G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio. 2017. “Graph Attention Networks.” arXiv preprint arXiv:1710.10903.

Vinayakumar, R., K. P. Soman, and P. Poornachandran. 2017. “Applying Deep Learning Approaches for Network Traffic Prediction.” 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI): 2353-2358, IEEE.

Wang, H., L. Liu, S. Dong, Z. Qian, and H. Wei. 2016. “A Novel Work Zone Short-Term Vehicle-Type Specific Traffic Speed Prediction Model Through the Hybrid EMD–ARIMA Framework.” *Transportmetrica B: Transport Dynamics* 4 (3): 159–186.

Wu, C. H., J. M. Ho, and D. T. Lee. 2004. “Travel-Time Prediction with Support Vector Regression.” *IEEE Transactions on Intelligent Transportation Systems* 5 (4): 276–281.

Xu, B., X. J. Ban, Y. Bian, W. Li, J. Wang, S. E. Li, and K. Li. 2018. “Cooperative Method of Traffic Signal Optimization and Speed Control of Connected Vehicles at Isolated Intersections.” *IEEE Transactions on Intelligent Transportation Systems* 20 (4): 1390–1403.

Yu, B., H. Yin, and Z. Zhu. 2017. “Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting.” arXiv preprint arXiv:1709.04875.

Yuan, J., Y. Zheng, X. Xie, and G. Sun. 2011. “Driving with Knowledge from the Physical World.” Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 316–324, ACM.

Zhan, X., S. Zhang, W. Y. Szeto, and X. Chen. 2019. “Multi-Step-Ahead Traffic Speed Forecasting Using Multi-Output Gradient Boosting Regression Tree.” *Journal of Intelligent Transportation Systems*, 1–17.

Zhang, Z., M. Li, X. Lin, Y. Wang, and F. He. 2019. “Multistep Speed Prediction on Traffic Networks: A Graph Convolutional Sequence-to-Sequence Learning Approach with Attention Mechanism.” *Transportation Research Part C: Emerging Technologies* 105: 297–322.

Zhou, Z., and X. Li. 2017. “Convolution on Graph: A High-Order and Adaptive Approach.” arXiv preprint arXiv:1706.09916.