Neurocomputing 461 (2021) 109–117

Contents lists available at ScienceDirect

Neurocomputing

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

Multiple dynamic graph based traffic speed prediction method

Zikai Zhang [a,b], Yidong Li [b,*], Haifeng Song [a], Hairong Dong [a,*]

[a] State Key Lab of Rail Traffic Control & Safety, Beijing Jiaotong University, Beijing 100044, China
[b] School of Computer and Information Engineering, Beijing Jiaotong University, Beijing 100044, China

ARTICLE INFO

Article history:
Received 25 April 2021
Revised 9 July 2021
Accepted 18 July 2021
Available online 22 July 2021
Communicated by Zidong Wang

Keywords:
Traffic prediction
Multiple graphs
Dynamic graph

ABSTRACT

Traffic speed prediction is a crucial and challenging task for intelligent transportation systems. The prediction task can be accomplished via graph neural networks with structured data, but accurate traffic speed prediction is challenging due to the complexity of traffic systems and the constantly dynamic changing nature. To address these issues, a novel evolution temporal graph convolutional network (ETGCN) model is proposed in this paper. The ETGCN model first fuses multiple graph structures, and utilizes graph convolutional network (GCN) to model spatial correlation. Then, the spatial–temporal dependence and their dynamical changes are learned simultaneously to predict traffic speed on a road network graph. Especially, a similarity-based attention method is proposed to fuse multiple graph adjacency matrices. Then, the gated recurrent unit is combined with GCN to capture spatial–temporal correlations and their changing status, simultaneously. Extensive experiments on two large-scale datasets show that our methods provide more accurate prediction results than the existing state-of-the-art methods in every prediction horizon.

1. Introduction

Due to the rapid growth of the urban population, the road is more crowded with vehicles, and thus result in environmental problems, traffic safety problems, and traffic congestion problems. In order to solve these problems, traffic speed prediction is gaining increasing attention from the public and the research community.

Traffic speed prediction aims to predict the future traffic speed in road networks based on historical observations (e.g., recorded via sensors). Recently, deep learning has shown its great learning and representation power in traffic speed prediction. To capture spatial correlation, researchers propose to use convolutional structures. For example, Zhang et al. [1] constructed three same convolutional structures to capture trend, period, and closeness information. Ma et al. [2] utilized CNN on the whole city for traffic speed prediction. Yao et al. [3] proposed a flow-gated local CNN to handle the spatial dependency by modeling the dynamic similarity among locations using traffic flow information. To model non-linear temporal dependency, researchers propose to use the recurrent neural network based framework. For example, Yu et al. [4] applied Long-short-term memory (LSTM) network to capture the sequential dependency for predicting the traffic under extreme conditions. Li et al. [5] utilized graph convolutional GRU for traffic speed prediction. To capture both long-term periodic information and temporal shifting, researchers propose to use weighted representation. For example, Yao et al. [3] proposed a periodically shifted attention mechanism by taking long-term periodic information and temporal shifting simultaneously. Wang et al. [6] used a Softmax Layer to get the weight vector of representation. Yu et al. [8] apply convolutional structures to extract spatio-temporal features simultaneously from graph-structured time series in a traffic study.

Non-Euclidean spatial correlations (such as content similarity) are also critical for accurate prediction. Intuitively, locations sharing similar functionality may have similar changing patterns. Similar regions may not necessarily be close in space. Therefore, find the location representing functional similarity (content similarity) among regions is helpful to enhance prediction accuracy. Only a few works attempt to deal with content similarity to enhance the prediction accuracy. Yuan et al. [7] treated the clustering stage as a separate sub-task and manually designed the distance measure, which is a non-trivial task. Wang et al. [6] proposed the end-to-end network to optimize the embedding parameters together with other parameters. The parameters are optimized through back-propagation towards minimizing the final prediction loss. For the multiple pair-wise correlations, Xu et al. [10] encode the non-Euclidean pair-wise correlations among regions into multiple graphs and then explicitly model these correlations using a

* Corresponding authors.
E-mail addresses: 18111064@bjtu.edu.cn (Z. Zhang), ydli@bjtu.edu.cn (Y. Li), songhf@bjtu.edu.cn (H. Song), hrdong@bjtu.edu.cn (H. Dong).

https://doi.org/10.1016/j.neucom.2021.07.052
0925-2312/

multi-graph convolution network (ST-MGCN). Yao et al. [9] propose a Deep Multi-View Spatial–Temporal Network (DMVST-Net) framework to model spatial, temporal, and semantic relations. But, these features are simply concatenated together, without a sufficient feature relationship extraction. To capture complex correlations, Yin et al. [24], propose a Multi-Stage Attention Spatial–Temporal Graph Networks to capture the interactions among multiple time series, to capture the spatial correlations within the same order neighborhood, to capture the spatial correlations among different neighborhoods, and to extract the dynamic temporal dependencies.

Accurate traffic prediction is very challenging mainly due to the following three complex factors: (1) Traffic data is generated by sensors, which are deployed along the road, as shown in Fig. 1(a). These data has no-linear and coupling relationships, which are usually effectively represented using graph structures. But, how to organize sensor nodes to form the graph is of great significance. (2) The representation and measurement of spatial relations are multiple, as shown in Fig. 1(b). These multiple spatial relations are reflected in geography correlation, region similarity, and road connectivity, which are usually fused to capture the traffic relationships. But, how to learn the traffic relationship for prediction with multiple spatial relations is challenging. (3) The spatial relations are dynamic changing over time, as shown in Fig. 1(c). How to capture the dynamically spatial–temporal correlations is important to improve the prediction ability.

To solve the above challenges, in this paper, we provide the **E**volution **T**emporal **G**raph **C**onvolutional **N**etwork (short as ETGCN) that captures the multiple spatial–temporal correlations and their dynamically changing status among nodes. Different from Xu, et al. [10], which capture correlations using a multi-graph convolution network, we propose the similarity-based graph adjacency matrix fusion method to explicitly model multiple spatial correlations together. Furthermore, to capture spatial–temporal relation and dynamic trends of traffic graph at the same time, we leverage the combined model that augments the GCN model with the gated recurrent unit (GRU) together to reweight the GCN parameter in different time slots. When evaluated on real-world datasets, ETGCN consistently outperforms state-of-the-art baselines by a large margin in every prediction horizon.

Our contributions are summarized as follows:

- We proposed three different kinds of graph adjacency matrices to identify spatial correlations. Then, we further fused the graph structures using the similarity-based graph adjacency matrix fusion method.
- We proposed an ETGCN model that captures spatial–temporal dependencies and their dynamic changes at the same time.
- We conducted extensive experiments on two large-scale datasets. The results show that our method outperforms the competing baselines in every prediction horizon.

The rest of the paper is organized as follows. We describe some notations and our problem in Section 2. The proposed method is presented in Section 3. We describe the experimental results in detail in Section 4. Finally, we conclude the paper in Section 5.

2. Related work

Traffic prediction plays an important role in the intelligent transportation system, which arised many statistic methods and machine learning methods. Earlier studies, such as ARIMA [28], VAR [29], SVM [27], bayesian methods [39], KNN [40], and artificial neural networks [41] are not satisfactory to handle complex traffic conditions.Deep learning based methods [44–46] are proposed to capture more variety of information for complex scenarios [42,43]. In order to improve the accuracy of traffic prediction,

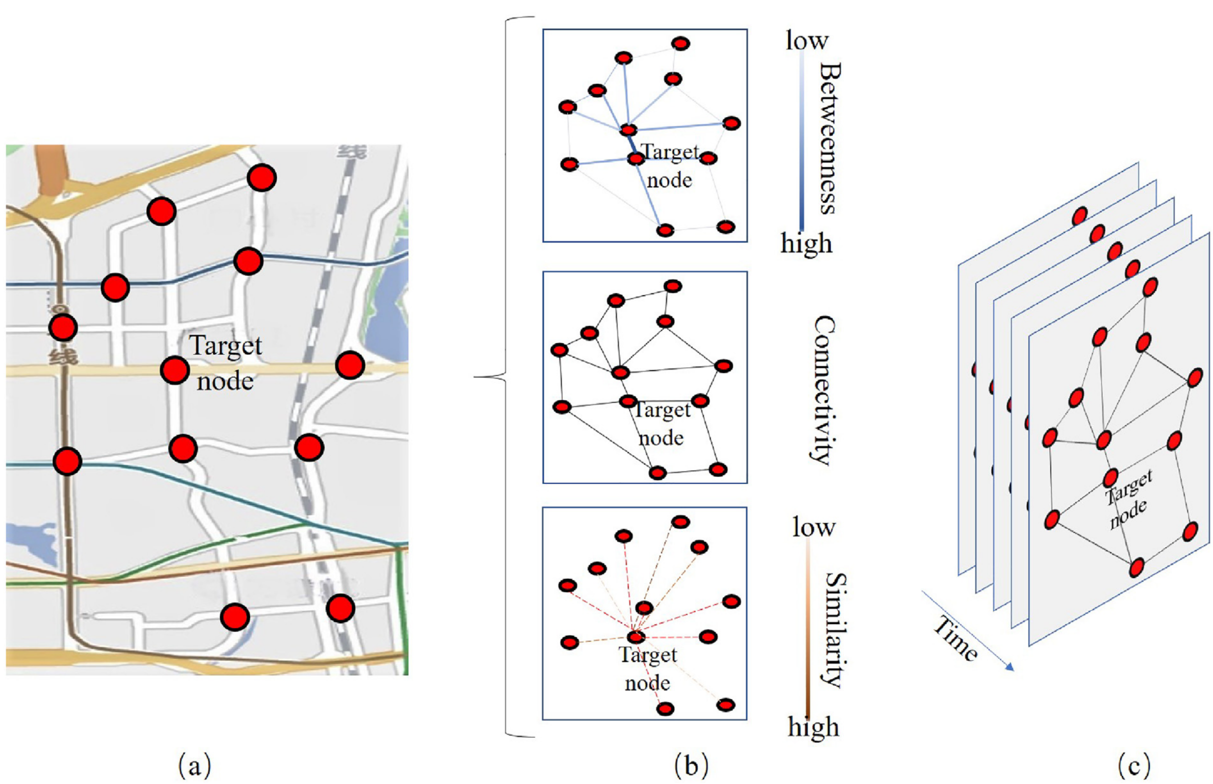

Fig. 1. (a) The perceived data are collected along the roadside; (b) There are multiple spatial relation representation methods; (c) The traffic graphs are dynamically evolving over time.

temporal relations, spatial relations, and semantic relations are captured in [32–35]. But, these methods are not suitable enough to model multiple relations and complex structural relations.

Recently, graph neural networks are widely used in traffic prediction tasks due to the spatial structural representation ability. Many methods have been proposed to solve traffic prediction problems through this technology. Diehl, et al. [30] interpret a traffic scene as a graph of interacting vehicles, and prove that interactions make GNNs worthwhile to traffic prediction systems. Li wei, et al. [26] propose MDCGCN to learn different period traffic patterns, and capture the dynamic temporal and spatial correlation caused by the changing relationship of traffic patterns among roads. J Ye, et al. [23] model the traffic system as a directed weighted graph and propose a spatio-temporal deep learning framework based on GCN and LSTM. What's more, the multi-range temporal relations and multi-distance spatial relations are considered and fused in [23] to have a better prediction result. X Shi, et al. [22] propose a novel Attention-based Periodic-Temporal neural Network (APTN), an end-to-end solution for traffic foresting that captures spatial, short-term, and long-term periodical dependencies with an encoder attention mechanism to model both the spatial and periodical dependencies. However, the above network models generally focus on a static graph, while the traffic graph is dynamically evolving over time in real-life traffic conditions.

One approach to solve this problem is combinations of graph based neural networks and recurrent architectures. To capture the spatial and temporal dependences simultaneously, L Zhang, et al. [18] propose a temporal graph convolutional network model, which is combined with the graph convolutional network (GCN) and the gated recurrent unit (GRU). Z. Li, et al. [19] propose a hybrid deep learning approach to efficiently capture the spatial–temporal features in traffic flow using GCN and LSTM. Guo, et al. [25] proposed a forecasting framework based on graph attention network (GAT) and temporal convolutional network (TCN) to capture spatial features and the temporal features. For a more accurate traffic prediction model, it is important to adapt to the dynamic changes of traffic conditions. Q song, et al. [20] propose GAC-Net (graph attention convolutional network) combining a novel temporal graph convolutional network and a spatio-temporal self-attention network to capture traffic status features and spatio-temporal features. However, spatial relations for different time slots are captured with shared graph convolutional kernel in above works, which may not efficient enough for the dynamic changing traffic modules.

For graph based traffic prediction methods, there are many ways to modeling the traffic graph. Z. Li, et al. [21] build a new graph by calculating the Spearman rank correlation coefficient between monitor stations and fine-tune the graph while training. Dai, et al. [31] prove that the correlation coefficient between the flow series of sensors (the semantic relations) is more important than the spatial location of sensors or regions. Aisan Kazerani, et al. [36] explore the betweenness centrality for analyzing traffic networks in traditional statistical ways. Alireza Ermagun, et al. [37] build network weight matrices that consist not simply of adjacency, but of network topology, network structure, and demand configuration. And, the network weight matrices operate better than traditional spatial graph matrices on traffic prediction. So, it is important to fuse different aspects of graphs for better traffic prediction learning. Guangyin Jin, et al. [38] propose a Deep Multi-View Spatiotemporal Virtual Graph Neural Network (DMVST-VGNN) with the correlation graph, the distance graph, and the mobility graph. However, the graph edge weights are $0 \setminus 1$ binary-valued and filtered out 90% correlations, which may not efficient enough to measure the spatial correlations. Moreover, the characteristics of road traffic network are not considered in [38] to improve the performance.

Motivated by the studies mentioned above, a novel network structure named Evolution Temporal Graph Convolutional Network (ETGCN) is proposed to predict traffic speed. Note that in the proposed method, the multiple graph structures are fused by a trainable similarity-based fusion method, and the GCN parameters are reweighted with GRU to capture the dynamically evolving spatial–temporal traffic relations for a better traffic prediction result.

3. Preliminaries

In this section, we fix some notations and define the traffic speed prediction problem.

Road Graph G**:** In this paper, we use a weighted graph $G = (V, A)$ to describe the topological structure of the road graph, and we treat each sensor as a node, where V is a set of all the sensor nodes, $V = \{v_1, v_2, \ldots, v_N\}, N$ is the number of the nodes. The adjacency matrix A is used to represent the dependency between nodes. The matrix A includes three types of information: content similarity adjacency matrix A_C, transportation betweenness adjacency matrix A_B and graph neighborhood adjacency matrix A_N.

Feature Matrix X**:** In this paper, we use the node attribute feature $X \in R^{M \times T}$ in the graph to describe the traffic speed on the transportation road network, where M represents the number of nodes in the graph and T represents the length of the historical time series. X_t is used to represents the speed on each node at time slot t.

Content Similarity Adjacency Matrix A_C**:** Content similarity indicates the similarity of historical time series between nodes. Cosine distance is suitable to measure the similarity of global temporal tends, since it is more about the difference in the direction of two vectors than the distance measure. So, in this paper, content similarity weight is characterized using cosine similarity for each pair of nodes. The content similarity adjacency matrix is defined as:

$$A_{C,ij} = \frac{X_i X_j^T}{|X_i||X_j|} \in [0, 1] \tag{1}$$

where $A_{C,ij}$ is the content similarity weight between two nodes v_i and v_j.

Transportation Neighborhood Adjacency Matrix A_N**:** Transportation neighborhood is defined based on the spatial proximity between nodes in the road graph. Transportation neighborhood weight could be characterized using normalized distance for each pair of connected nodes. In [38], the correlation is measured with $\frac{1}{d_{ij}}$, and reweight the top 10% connections with 1. In this paper, we want to measure the Matrix A_N with a soft weight in a similar way, and propose the transportation neighborhood adjacency matrix as:

$$A_{N,ij} = 1 - \frac{d_{ij}}{max(D)} \in [0, 1] \tag{2}$$

where $A_{N,ij}$ is the transportation neighborhood weight between two nodes v_i and v_j, d_ij is the distance between v_i and v_j, D is the set of distance in the transportation road network, and the function $max()$ is used to find the maximum distance in the transportation road network.

Graph Betweenness Adjacency Matrix A_B**:** Graph betweenness is defined based on the edge betweenness centrality of each pair of nodes in the road graph. Graph betweenness weight could be characterized using the sum of the fraction of all-pairs shortest paths that pass through the edge (a pair of nodes), and the graph betweenness adjacency matrix is defined as:

$$A_{B,ij} = \sum_{k,w \in V} \frac{\delta(k,w|i,j)}{\delta(k,w)} \in [0,1] \quad (3)$$

where $A_{B,ij}$ is the set of nodes, $\delta(k,w)$ is the number of shortest (k,w)-paths, and $\delta(k,w|i,j)$ is the number of those paths passing through edge between v_i and v_j.

Traffic Speed Prediction Problem: In this paper, the goal of traffic prediction is to predict traffic speed in a certain period of time based on the historical traffic information on the roads. Given road graph G and the previous T input speeds ($X = [X_{t-T+1}, \ldots, X_t]$), we want to learn a function $f()$ that maps historical speeds of all nodes to the speed in the next timestep. The prediction equation is as follows,

$$X_{t+1} = f(X_{t-T+1}, \ldots, X_t; G). \quad (4)$$

4. Methods

In this section, we provide Evolution Temporal Graph Convolutional Network (ETGCN) that captures the spatial correlation, temporal correlation, and their dynamic change status among nodes in the graph. The ETGCN consists of two important parts. The first one is multiple adjacency matrix fusion, which is used to fuse different aspects of adjacency matrices for better feature learning. The second component is the ETGCN cell, which is used to learn the spatial feature and temporal feature by using a recurrent model to evolve the GCN model. Based on the ETGCN cell, ETGCN is easily used to learn the spatial features in depth and temporal features in sequence.

4.1. Multiple adjacency matrix fusion

In this paper, the transportation graph adjacency matrices are constructed as the content similarity adjacency matrix A_C, the transportation neighborhood adjacency matrix A_N, and the graph betweenness adjacency matrix A_B. Motivated by the similarity-based attention mechanism, we propose the multiple adjacency matrix fusion method to learn the spatial dependency between nodes in the graph as follows,

$$A_F = W_3 * (W_1 * A_B \times A_N + W_2 * A_B \times A_C) \times A_B \quad (5)$$

where $*$ denote the dot product and $\times$ denote the cross product. $A_F \in R^{M \times M}$ is the final fused graph adjacency matrix. $W_1 \in R^{M \times M}$ and $W_2 \in R^{M \times M}$ are parameters to balance the influence of A_N and A_C. $W_3 \in R^{M \times M}$ is the parameter to adjust the final matrix.

In the fusion process, we first apply the cross-product to obtain the similarity score between graph matrices. Usually, the similarity is measured with dot-product, concatenate, or perceptron. But, the similarity between graph matrices is measured with cross-product to take advantage of graph structure relations. For example, given cross-product equation $\{A \times B\}_{ij} = A_i B_j^T$, we can obtain the graph fusion matrix $A \times B$. A_i means the message passing rate from node i to other nodes, and B_j^T means the message passing rate from other nodes to node j. $A_i B_j^T$ means the fused message passing rate from node i to node j.

As shown in Fig. 2, we first obtain the graph fusion matrix $A_B \times A_N$ and $A_B \times A_C$. Then, using W_1 and W_2, we further fuse the graph fusion matrices. After regularization operation and cross-product operation, we get the final fused graph adjacency matrix A_F.

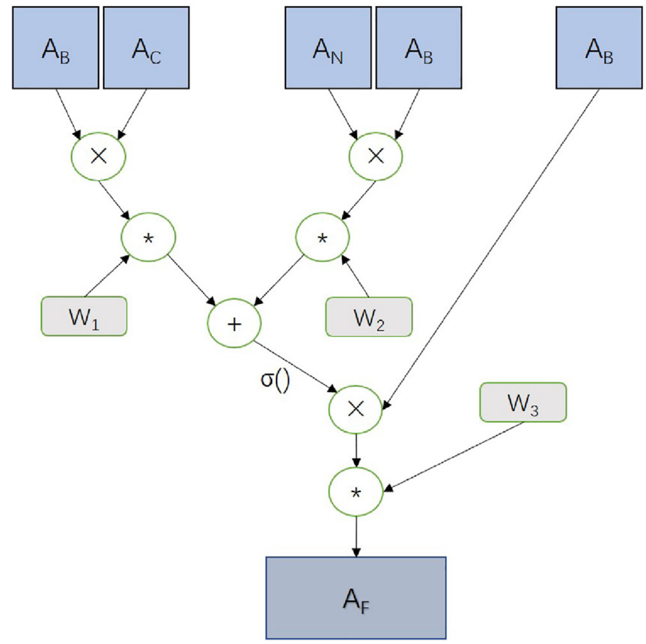

Fig. 2. Graph matrix fusion.

4.2. Evolving graph convolution

4.2.1. Graph Convolutional Network (GCN)

The spatial dependence is important in traffic speed prediction. Previously, the convolutional neural network (CNN) is used to obtain local spatial features from Euclidean domains data which have regular spatial structure. But, most data in traffic scenarios have irregular spatial structures, where each node in these graphs has different connections. It is difficult for CNN to accurately capture spatial dependence on the traffic road network. Recently, the graph convolutional network (GCN) has received widespread attention and successfully used in traffic prediction/forecasting.

To capture the complex spatial dependence in traffic prediction problems, the GCN model constructs a filter in the Fourier domain to capture spatial features between the nodes. With Fourier transform, the spectral domain on the traffic road network is modified to regular spatial structure. The GCN model can be built by stacking multiple convolutional layers, and the recursive equation is as follows,

$$H^{l+1} = f(H^l, A) = \sigma(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^l W^l) \quad (6)$$

$$H^0 = X \quad (7)$$

where the l-th layer takes input H^l with parameter $W^l, \tilde{A} = A + I$, and the augmented diagonal degree of graph G is $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$.

4.2.2. Gated Recurrent Unit (GRU)

To capture the complex both long-term and short-term temporal dependencies in the traffic prediction problem, the GRU model presents the advantages of using gated neurons to catch both the short-term and the long-term memories while avoiding the gradient vanishing problem. The GRU models can obtain the comparable results with the recursive equation as follows,

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \quad (8)$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \quad (9)$$

$$\tilde{h}_t = tanh(W \cdot [r_t * h_{t-1}, x_t]) \quad (10)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \quad (11)$$

where the reset gate and update gate at time slot t are denoted as r_t and z_t, respectively. h_{t-1} and h_t denote the hidden state at time slot

$t-1$ and t, respectively. x_t is the input at time slot t, which is the same as the past output h_{t-1}.

4.2.3. Evolving Temporal GCN (ETGCN)

Motivated by EvolveGCN [11], we propose the ETGCN combining the GCN, the GRU, and the multiple adjacency matrix fusion method to capture the spatial and temporal dependences at the same time. The ETGCN model captures the multiple spatial correlations and captures the dynamism of the graph sequence through the ETGCN cell.

The specific calculation process of an ETGCN cell is shown in Fig. 3. U_t and R_t denote the reset gate matrix and the update gate matrix at time slot t, respectively. Z_{t-1} denotes the GCN parameter at time $t-1$, and X_t denotes the input feature at time t. Z_t denotes the GCN parameter at time t, and A_F denotes the final fused graph adjacency matrix. As shown in Fig. 3, ETGCN first learns the GCN parameter using the GRU technique. Then combined with the fused graph adjacency matrix, the spatial–temporal feature is learned using GCN. The ETGCN equation is as follows,

$$U_t = \sigma(W_u \cdot Z_{t-1} + D_u \cdot X_t) \quad (12)$$

$$R_t = \sigma(W_r \cdot Z_{t-1} + D_r \cdot X_t) \quad (13)$$

$$C_t = tanh(W_c \cdot (R_t * Z_{t-1}) + D_c \cdot X_t) \quad (14)$$

$$Z_t = (1 - U_t) * Z_{t-1} + U_t * C_t \quad (15)$$

$$Y_t = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}_F\tilde{D}^{-\frac{1}{2}}X_tZ_t) \quad (16)$$

As shown in Fig. 4, the ETGCN model is made up of ETGCN cells. The ETGCN model can deal with complex spatial dependence and temporal dynamics. The ETGCN model is different from other spatial–temporal models in two points. On one hand, compared with other models, the ETGCN model has good expansibility. The depth of network layers for spatial feature learning and the length of temporal feature learning are easy to expand. On the other hand, the ETGCN model has a better adaptive ability with the change of time. At each time slot, the time series model (GRU/LSTM) is used to optimize the parameters of spatial feature learning. Spatial features can be dynamically updated with GCN parameters changing over time.

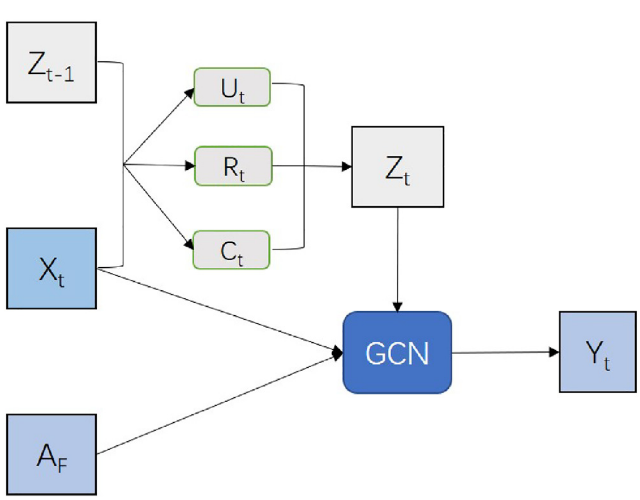

Fig. 3. ETGCN cell.

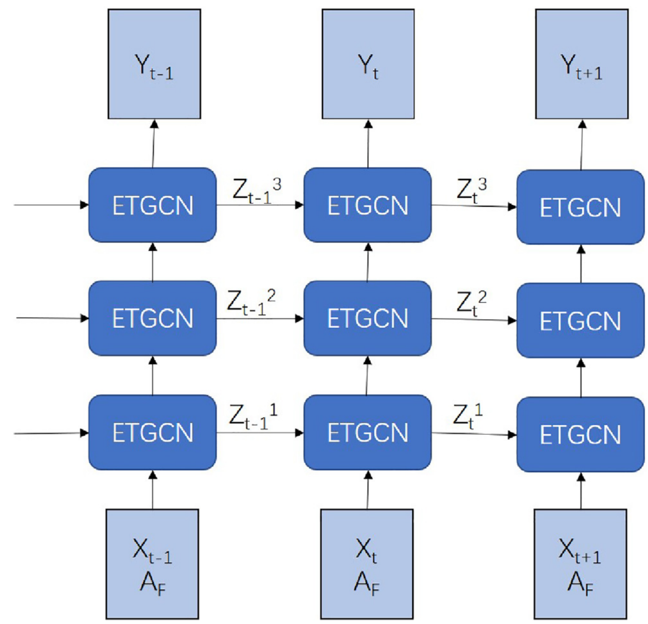

Fig. 4. ETGCN.

4.3. Computational complexity analysis

To analysis the computational complexity of the proposed method, we divide our method into 2 components, i.e., the ETGCN network and the optimization method (SGD method in this paper).

Suppose the number of time slots is L, the dimension of feature is K, and the number of graph nodes is M. Both the time complexity and the space complexity of the A_F generation process are $O(M^2)$ for one epoch of training, according to formula (5). The computational complexity of the normal GRU method [47] is $O(d^2)$, and d is its embedding dimension. Then, according to the formula (12–16), the time complexity of the ETGCN cell can be calculated as $O(M^6(M^2)) = O(M^8)$, and the space complexity of the ETGCN cell is $O(M^2)$ at each epoch training step ($M \gg K$ in this paper). According to Fig. 4, the time complexity of the ETGCN network is $O(L(M^8))$, and the space complexity of the ETGCN is $O(L(M^2))$.

Suppose the accuracy (approximation error) of the SGD objective function is ϵ, the batch size of samples is B, the iterations of SGD is $O(1/\epsilon)$ [48], and the space complexity of SGD is $O(K)$. So, the overall time complexity of the proposed method is $O(LM^8 \times 1/\epsilon) = O(LM^8/\epsilon)$, and the overall space complexity of the proposed method is $O(B \times (L(M^2) + K)) = O(BLM^2)$.

5. Experiments and analysis

In this section, we conduct experiments on the real datasets: the SZ-taxi dataset and the xiamen dataset. We show a comprehensive quantitative evaluation compared with other methods and also show the effectiveness of the ETGCN.

5.1. Datasets and training

SZ-taxi dataset [10] is the taxi trajectory of Shenzhen with 156 major roads of Luohu District as the study area from Jan. 1 to Jan. 31, 2015. Xiamen dataset [17] contains 5 months of data recorded by 95 traffic sensors ranging from August 1st, 2015 to December 31st, 2015 in Xiamen, China.

In our experiments, for training, validation, and testing, we divide the data into three data sets by the ratio of 8:1:1 according to the time interval. When training and testing, the basic time slot is set as 5 min. We use the previous 12 time slots (1 h) data to predict the result.

5.2. Evaluation metrics

In our experiment, we use Mean Average Percentage Error (MAPE), Mean Absolute Error (MAE), Accuracy value, and the coefficient of determination(R^2) score as the evaluation metrics. These metrics are defined as follows,

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|\hat{y}^i_{T+1} - y^i_{T+1}\right| \tag{17}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\hat{y}^i_{T+1} - y^i_{T+1}\right)^2} \tag{18}$$

$$Accuracy = 1 - \frac{1}{N}\sum_{i=1}^{N}\frac{\left|\hat{y}^i_{T+1} - y^i_{T+1}\right|}{\left|y^i_{T+1}\right|} \tag{19}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\left(\hat{y}^i_{T+1} - y^i_{T+1}\right)^2}{\sum_{i=1}^{N}\left(y^i_{T+1} - \bar{y}_{T+1}\right)^2} \tag{20}$$

where $\hat{y}^i_{T+1}$ and y^i_{T+1} represent the prediction value and the real value of region l_i for time interval $T+1$, $\bar{y}_{T+1}$ represent the average real value for time interval $T+1$, and N is the total number of samples.

5.3. Compared algorithms

We compare the proposed ETGCN with the following baseline methods:

(1) ARIMA[12]: Autoregressive Integrated Moving Average model (ARIMA) considers moving average and autoregressive components to predict traffic data.
(2) ISVR [13]: ISVR is an incremental support vector regression method to predict time-serial data.
(3) LSTM [14]: Long short-term memory (LSTM) is one kind of recurrent neural network to learn the spatial feature of sequence data.
(4) GCN [15]: Graph convolutional network (GCN) is used to learn spatial features for all time slots from traffic data.
(5) T-GCN [10]:T-GCN models the spatio-temporal dependency by integrating graph convolution into the gated recurrent unit.
(6) ST-MGCN [16]: Spatiotemporal multi-graph convolution network (ST-MGCN) encode the pair-wise correlations among regions into multiple graphs and then explicitly model these correlations using multi-graph convolution.
(7) GMAN [17]: GMAN is a graph multi-attention network, which utilizes spatial and temporal attention mechanisms with gated fusion to model the complex spatio-temporal correlations.

5.4. Performance

Tables 1 and 2 show the ETGCN model and other baseline methods for 15 min, 30 min, 45 min, and 60 min on the datasets. It can be seen that the ETGCN model obtains the best prediction performance under all evaluation metrics for all prediction horizons.

(1) Spatio-temporal based deep learning methods, including T-GCN, ST-MGCN, GMAN, and the proposed ETGCN, which are able to model the non-linear spatio-temporal dependencies, generally outperform other baselines. For example, for the 15-min traffic prediction task, the MAE errors and the RMSE errors of the ARIMA, ISVR, GCN, and LSTM models are larger than the MAE errors and the RMSE errors of the T-GCN, ST-MGCN, GMAN, ETGCN. The accuracy score and the R^2 score of the ARIMA, ISVR, GCN, and LSTM models are smaller than the accuracy score of the T-GCN, ST-MGCN, GMAN, ETGCN. This is mainly due to methods such as the ARIMA, ISVR, GCN, and LSTM that find it difficult to handle complex temporal and spatial data.
(2) Among deep learning methods, multi-attention-based graph models(including ST-MGCN and GMAN) and the proposed ETGCN perform better than T-GCN, indicating the multiple edge relations are helpful to capture graph features for traffic prediction. For example, for the 15-min traffic prediction task, the MAE errors, and the RMSE errors of the ST-MGCN, GMAN, and ETGCN models are smaller than the MAE errors and the RMSE errors of the T-GCN. The accuracy score andthe R^2 score of the ST-MGCN, GMAN, and ETGCN models are larger than the accuracy score of the T-GCN.

Table 1
The prediction results of the ETGCN model and other baseline methods on SZ-taxi dataset.

T	Metrics	Methods							
		ARIMA	LSTM	ISVR	GCN	T-GCN	ST-MGCN	GMAN	ETGCN
15 min	MAE	18.659	13.407	14.781	17.425	11.636	10.790	10.757	10.631
	RMSE	24.645	20.242	22.610	22.252	16.840	16.506	15.273	14.624
	Accuracy	0.428	0.718	0.696	0.643	0.729	0.731	0.742	0.751
	R^2	0.687	0.789	0.737	0.745	0.854	0.860	0.880	0.890
30 min	MAE	18.643	13.505	14.946	17.571	11.804	11.779	11.307	11.006
	RMSE	24.640	20.385	22.424	22.428	17.035	16.574	15.411	15.635
	Accuracy	0.428	0.716	0.699	0.641	0.726	0.727	0.738	0.744
	R^2	0.677	0.779	0.732	0.732	0.846	0.854	0.874	0.870
45 min	MAE	18.646	13.604	15.100	17.689	11.896	11.490	11.453	11.734
	RMSE	24.640	20.501	22.427	22.566	17.179	16.710	16.843	15.702
	Accuracy	0.428	0.714	0.699	0.638	0.714	0.723	0.724	0.733
	R^2	0.679	0.778	0.734	0.731	0.844	0.852	0.850	0.870
60 min	MAE	18.635	13.716	15.214	17.789	11.992	11.515	12.184	11.387
	RMSE	24.619	20.621	22.465	22.681	17.261	16.841	17.013	15.723
	Accuracy	0.428	0.713	0.698	0.637	0.713	0.723	0.719	0.732
	R^2	0.679	0.775	0.733	0.728	0.842	0.850	0.847	0.869

Table 2
The prediction results of the ETGCN model and other baseline methods on xiamen dataset.

T	Metrics	Methods							
		ARIMA	LSTM	ISVR	GCN	T-GCN	ST-MGCN	GMAN	ETGCN
15 min	MAE	14.819	12.513	13.059	17.425	11.766	11.407	11.500	11.777
	RMSE	25.033	20.791	21.470	22.252	19.944	18.498	19.525	15.394
	Accuracy	0.576	0.694	0.636	0.623	0.707	0.732	0.713	0.738
	R^2	0.785	0.851	0.842	0.830	0.863	0.882	0.869	0.919
30 min	MAE	18.834	13.743	15.665	17.571	13.199	12.794	12.027	12.195
	RMSE	33.092	23.933	26.342	22.428	23.294	21.660	21.427	16.083
	Accuracy	0.520	0.648	0.554	0.620	0.658	0.686	0.685	0.726
	R^2	0.637	0.810	0.770	0.833	0.820	0.845	0.848	0.914
45 min	MAE	22.630	14.951	0.629	17.696	14.426	13.352	12.375	12.760
	RMSE	40.190	26.274	32.077	22.510	26.416	23.161	22.823	16.333
	Accuracy	0.4175	0.619	0.535	0.619	0.617	0.664	0.669	0.722
	R^2	0.490	0.782	0.675	0.840	0.780	0.831	0.835	0.916
60 min	MAE	26.589	16.021	20.698	17.789	15.836	13.793	12.792	11.937
	RMSE	46.320	29.570	35.864	22.681	29.412	27.569	24.150	16.384
	Accuracy	0.215	0.565	0.392	0.616	0.568	0.601	0.645	0.721
	R^2	0.315	0.721	0.589	0.836	0.724	0.757	0.814	0.914

(3) ETGCN achieves state-of-the-art prediction performances and the advantages are more evident in the long-term traffic prediction task. For example, In Table 1, the accuracy score of the proposed ETGCN is better than ST-MGCN and GMAN by 3.22% and 1.69% for the 15-min traffic prediction task. The accuracy score of the proposed ETGCN is better than ST-MGCN and GMAN by 2.37% and 0.79% for the 30-min traffic prediction task. The accuracy score of the proposed ETGCN is better than ST-MGCN and GMAN by 1.26% and 1.19% for the 45-min traffic prediction task. The accuracy score of the proposed ETGCN is better than ST-MGCN and GMAN by 1.29% and 1.84% for the 60-min traffic prediction task.

5.5. Effect of graph matrix fusion and ETGCN cell

In this subsection, one group of ablation studies are conducted to evaluate contributions of the ETGCN method and the graph matrix fusion method.

We first ablate the graph matrix fusion method from ETGCN model, and denoted as ETGCN- model. Then, changing the ETGCN cell to GRU cell, the ETGCN- model is degraded to the T-GCN model.

As shown in Fig. 5, compared with the ETGCN- model, the RMSE error and the MAE error of the ETGCN model are reduced by 4.99% and 4.70%, respectively. The ETGCN model performs better than the ETGCN- model, which shows the effectiveness of the graph matrix fusion method. Compared with the T-GCN model, the RMSE error and the MAE error of the ETGCN- model are reduced by 8.59% and 4.13%, respectively. The ETGCN- model performs better than the T-GCN model, which shows the effectiveness of the ETGCN cell.

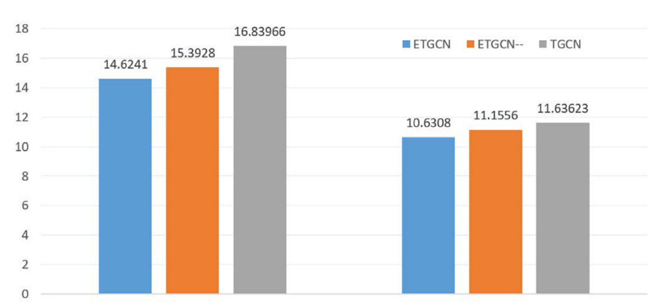

Fig. 5. Ablation study about graph matrix fusion and ETGCN cell.

5.6. Effect of different graph matrix fusion methods

In this subsection, two groups of matrix fusion methods are conducted to evaluate the effect of different graph matrix fusion methods.

In the ETGCN model, the graph matrix fusion equation is $A_F = W_3 * (W_1 * A_B \times A_N + W_2 * A_B \times A_C) \times A_B$. To evaluate the effect of the cross-product on graph matrix fusion, we build for the normal matrix fusion method without using cross-product operation. In ETGCN-cbn model, the graph matrix fusion equation is $A_F = W_1 * A_N + W_2 * A_C + W_3 * A_B$. In the ETGCN-cb model, the graph matrix fusion equation is $A_F = W_1 * A_C + W_2 * A_B$. In the ETGCN-cn model, the graph matrix fusion equation is $A_F = W_1 * A_N + W_2 * A_C$. In ETGCN-bn model, the graph matrix fusion equation is $A_F = W_1 * A_N + W_2 * A_B$.

As shown in Table 3, The ETGCN model performs best compared with other models, which shows the effectiveness of the cross-product on graph matrix fusion. The ETGCN model performs better than the ETGCN- model, which shows the effectiveness of the graph matrix fusion method.

To further explore the effectiveness of the graph matrix fusion method, we build another group of matrix fusion methods.

In ETGCN model, the graph matrix fusion equation is $A_F = W_3 * (W_1 * A_B \times A_N + W_2 * A_B \times A_C) \times A_B$. In ETGCN-2 model, the graph matrix fusion equation is $A_F = W_3 * [(W_1 * A_B \times A_N) * (W_2 * A_B \times A_C)] \times A_B$. In ETGCN-3 model, the graph matrix fusion equation is $A_F = W_3 * (W_1 * A_B \times A_N + W_2 * A_B \times A_C) * A_B$. In ETGCN-4 model, the graph matrix fusion equation is $A_F = W_3 * [(W_1 * A_B \times A_N) * (W_2 * A_B \times A_C)] * A_B$.

Table 4 shows the group of matrix fusion methods for 15 min, 30 min, 45 min, and 60 min. From Table 4, We can get two observations. (1) The ETGCN model obtains better prediction performance than the ETGCN-2 model for all prediction horizons. The

Table 3
Effect of different graph matrix fusion methods

MODEL	Metrics		
	MAE	RMSE	Accuracy
ETGCN	10.631	14.624	0.751
ETGCN-cbn	11.193	15.785	0.731
ETGCN-cb	11.154	15.109	0.743
ETGCN-cn	11.050	15.212	0.741
ETGCN-bn	11.779	16.074	0.726

Table 4
Comparison with different graph matrix fusion methods

T	Metrics	Methods			
		ETGCN	ETGCN-2	ETGCN-3	ETGCN-4
15 min	MAE	10.631	11.025	10.541	11.530
	RMSE	14.624	14.756	14.973	15.527
	Accuracy	0.751	0.749	0.745	0.736
30 min	MAE	11.005	12.079	10.553	11.331
	RMSE	15.635	17.771	14.953	15.205
	Accuracy	0.744	0.731	0.745	0.741
45 min	MAE	11.734	11.654	10.776	11.357
	RMSE	15.702	15.822	15.416	16.090
	Accuracy	0.733	0.731	0.737	0.726
60 min	MAE	11.387	11.863	11.652	12.412
	RMSE	15.723	16.117	16.051	16.653
	Accuracy	0.732	0.725	0.728	0.716

ETGCN-3 model obtains better prediction performance than the ETGCN-4 model for all prediction horizons. (2) The ETGCN model obtains better prediction performance than the ETGCN-3 model for 15 min and 60 min prediction tasks. But, the ETGCN-3 model obtains better prediction performance than the ETGCN model for 30 min and 45 min prediction tasks. The ETGCN model obtains the top two performances among the group of methods for all prediction horizons.

6. Conclusions

In this paper, we propose a traffic speed prediction model, named ETGCN, based on spatial–temporal feature learning. The model fused the multiple graph structures to learn the spatial correlations using a similarity-based fusion method and graph convolution network. Furthermore, the spatial–temporal dependencies and their dynamically changing status are captured simultaneously, using the gated recurrent unit combined with a graph convolutional network. The evaluations show that our proposed model significantly outperforms the state-of-the-art methods on real-world datasets.

CRediT authorship contribution statement

Zikai Zhang: Conceptualization, Methodology, Validation, Writing - original draft. **Yidong Li:** Writing - review & editing, Funding acquisition, Supervision. **Haifeng Song:** Supervision, Project administration. **Hairong Dong:** Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Joint Funds of the National Natural Science Foundation of China under Grant No. U1934220, the National Science Fund for Distinguished Young Scholars under Grant No. 61925302, and the National Science Foundation of China under No. 61903021.

References

[1] Junbo Zhang, Yu Zheng, Dekang Qi, "Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction. AAAI Conference on Artificial Intelligence, AAAI (2017), 2017..
[2] Xiaolei Ma, Zhuang Dai, Zhengbing He, Jihui Ma, Yong Wang, Yunpeng Wang "Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction, Sensors 17(4) (2017) 818..
[3] H. Yao, X. Tang, H. Wei, et al., Revisiting spatial-temporal similarity: a deep learning framework for traffic prediction, in: AAAI Conference on Artificial Intelligence, AAAI, 2019.
[4] Yu. Rose, Yaguang Li, Ugur Demiryurek, Cyrus Shahabi, Yan Liu, Deep learning: a generic approach for extreme condition traffic forecasting, in: Proceedings of SIAM International Conference on Data Mining, 2017.
[5] Yaguang Li, Yu. Rose, Cyrus Shahabi, Yan Liu, Diffusion convolutional recurrent neural network: data-driven traffic forecasting, in: Sixth International Conference on Learning Representations, 2018.
[6] D. Wang, W. Cao, J. Li, et al., DeepSD: Supply-demand prediction for online car-hailing services using deep neural networks, in: 2017 IEEE 33rd International Conference on Data Engineering (ICDE), IEEE Computer Society, 2017.
[7] N.J. Yuan, Y. Zheng, L. Zhang, X. Xie, T-finder: A recommender system for finding passengers and vacant taxis, IEEE Transactions on Knowledge and Data Engineering 25 (10) (2013) 2390–2403.
[8] B. Yu, H. Yin, Z. Zhu, Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting, in: IJCAI'18, 2018.
[9] Huaxiu Yao, Fei Wu, Jintao Ke, Xianfeng Tang, Yitian Jia, Siyu Lu, Pinghua Gong, Jieping Ye, Zhenhui Li, "Deep Multi-View Spatial-Temporal Network for Taxi Demand Prediction. AAAI Conference on Artificial Intelligence (2018), 2018..
[10] Xu. Geng et al., Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting, in: AAAI Conference on Artificial Intelligence, 2019.
[11] Aldo Pareja et al., EvolveGCN: Evolving graph convolutional networks for dynamic graphs, in: AAAI Conference on Artificial Intelligence, 2020.
[12] Dimitros Asteriou, Stephen G Hall, Arima models and the box-jenkins methodology, Applied Econometrics 2 (2) (2011) 265–286.
[13] Su, Haowei, L. Zhang, S. Yu. "Short-term traffic flow prediction based on incremental support vector regression, Natural Computation, 2007. ICNC 2007. Third International Conference on IEEE Computer Society, 2007..
[14] Yongxue Tian, L. Pan, Predicting short-term traffic flow by long short-term memory recurrent neural network, IEEE International Conference on Smart City/socialcom/sustaincom, IEEE (2015).
[15] Thomas N. Kipf, Max Welling, Semi-supervised classification with graph convolutional networks, in: ICLR 2017: International Conference on Learning Representations 2017, 2017.
[16] Geng, Xu, et al, "Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting, AAAI 2019: Thirty-Third AAAI Conference on Artificial Intelligence, vol. 33, no. 1, 2019, pp. 3656–3663..
[17] Zheng, Chuanpan, et al. "GMAN: A Graph Multi-Attention Network for Traffic Prediction. AAAI 2020: The Thirty-Fourth AAAI Conference on Artificial Intelligence, vol. 34, no. 1, 2020, pp. 1234–1241..
[18] L. Zhao et al., T-GCN: A temporal graph convolutional network for traffic prediction, IEEE Transactions on Intelligent Transportation Systems 21 (9) (Sept. 2020) 3848–3858, https://doi.org/10.1109/TITS.2019.2935152.
[19] Z. Li et al., A hybrid deep learning approach with GCN and LSTM for traffic flow prediction*, 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 2019, pp. 1929-1933, doi: 10.1109/ITSC.2019.8916778..
[20] Q. Song, R. Ming, J. Hu, H. Niu M. Gao, Graph Attention convolutional network: spatiotemporal modeling for urban traffic prediction, 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 2020, pp. 1-6, doi: 10.1109/ITSC45102.2020.9294580..
[21] Z. Li et al., A multi-stream feature fusion approach for traffic prediction, in: IEEE Transactions on Intelligent Transportation Systems, doi: 10.1109/TITS.2020.3026836..
[22] X. Shi, H. Qi, Y. Shen, G. Wu B. Yin, A spatial-temporal attention approach for traffic prediction, in: IEEE Transactions on Intelligent Transportation Systems, doi: 10.1109/TITS.2020.2983651..

[23] J. Ye, J. Zhao, K. Ye, C. Xu, Multi-STGCnet: A graph convolution based spatial-temporal framework for subway passenger flow forecasting, 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 2020, pp. 1–8, doi: 10.1109/IJCNN48605.2020.9207049..
[24] Xueyan Yin et al., Multi-stage attention spatial-temporal graph networks for traffic prediction, Neurocomputing 428 (2021) 42–53.
[25] Ge Guo, Wei Yuan, Short-term traffic speed forecasting based on graph attention temporal convolutional networks, Neurocomputing 410 (2020) 387–393.
[26] Wei Li et al., Traffic flow prediction over muti-sensor data correlation with graph convolution network, Neurocomputing 427 (2021) 50–63.
[27] Pvvk Theja, L. Vanajakshi, Short term prediction of traffic parameters using support vector machines technique, in: International Conference on Emerging Trends in Engineering & Technology IEEE, 2011.
[28] S. Vasantha Kumar, Lelitha Vanajakshi, Short-term traffic flow prediction using seasonal arima model with limited input data, European Transport Research Review 7 (3) (2015) 1–9.
[29] F. Schimbinschi, et al., Topology-regularized universal vector autoregression for traffic forecasting in large urban areas. Expert Systems with Application 82 (Oct.) (2017) 301–316..
[30] Frederik Diehl et al., Graph neural networks for modelling traffic participant interaction, in: 2019 IEEE Intelligent Vehicles Symposium (IV), 2019, pp. 695–701.
[31] Xingyuan Dai et al., DeepTrend 2.0: A light-weighted multi-scale traffic prediction model using detrending, Transportation Research Part C-Emerging Technologies 103 (2019) 142–157.
[32] Zhixiang He et al., STCNN: A spatio-temporal convolutional neural network for long-term traffic prediction, in: 2019 20th IEEE International Conference on Mobile Data Management (MDM), 2019, pp. 226–233.
[33] Xiangdong Ran et al., Short-term travel time prediction: a spatiotemporal deep learning approach, International Journal of Information Technology and Decision Making 18 (4) (2019) 1087–1111.
[34] Shengjian Zhao et al., Time series traffic prediction via hybrid neural networks, in: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 2019, pp. 1671–1676.
[35] Lingyi Han et al., Short-term traffic prediction based on deepcluster in large-scale road networks, IEEE Transactions on Vehicular Technology 68 (12) (2019) 12301–12313.
[36] A. Kazerani, S. Winter, Modified betweenness centrality for predicting traffic flow, in: Proceedings of the 10th International Conference on GeoComputation, 2009.
[37] Alireza Ermagun, David M. Levinson, Development and application of the network weight matrix to predict traffic flow for congested and uncongested conditions, Environment and Planning B: Urban Analytics and City Science 46 (9) (2019) 1684–1705.
[38] Guangyin Jin et al., Deep multi-view spatiotemporal virtual graph neural network for significant citywide ride-hailing demand prediction, ArXiv Preprint ArXiv:2007.15189 (2020).
[39] Bradford S. Westgate et al., Travel time estimation for ambulances using bayesian data augmentation, The Annals of Applied Statistics 7 (2) (2013) 1139–1161.
[40] Gary A. Davis, Nancy L. Nihan, Nonparametric regression and short term freeway traffic forecasting, Journal of Transportation Engineering-Asce 117 (2) (1991) 178–188.
[41] Fabio Moretti et al., Urban traffic flow forecasting through statistical and neural network bagging ensemble hybrid modeling, Neurocomputing 167 (2015) 3–7.
[42] Wenju Du, Yinzhen Li, Jiangang Zhang, Yu Jianning, Synchronisation between two different networks with multi-weights and its application in public traffic network, International Journal of Systems Science 50 (3) (2019) 534–545.
[43] Zhuoyang Du, Dong Li, Kaiyu Zheng, Shan Liu, Speed profile optimisation for intelligent vehicles in dynamic traffic scenarios, International Journal of Systems Science 51 (12) (2020) 2167–2180.
[44] Nianyin Zeng et al., Facial expression recognition via learning deep sparse autoencoders, Neurocomputing 273 (2018) 643–649.
[45] Nianyin Zeng et al., An improved particle filter with a novel hybrid proposal distribution for quantitative analysis of gold immunochromatographic strips, IEEE Transactions on Nanotechnology 18 (2019) 819–829.
[46] Nianyin Zeng et al., Deep-reinforcement-learning-based images segmentation for quantitative analysis of gold immunochromatographic strip, Neurocomputing 425 (2021) 173–180.
[47] B. Zhang, D. Xiong, J. Xie, J. Su, Neural machine translation with GRU-gated attention model, IEEE Transactions on Neural Networks and Learning Systems 31 (11) (2020) 4688–4698.
[48] Liang, Shiyu, R. Srikant, Why deep neural networks for function approximation, 5th International Conference on Learning Representations, ICLR 2017, 2016..

Zikai Zhang received the B.S. degree from Tianjin polytechnic University, China, in 2014. he is currently pursuing the Ph.d. degree with School of Electronic and Information Engineering and the State Key Lab of Rail Traffic Control & Safety, Beijing Jiaotong University, China. His main research interests lie in big data analysis, security and intelligent transportation, and deep learning

Yidong Li is a professor in the School of Computer and Information Technology at Beijing Jiaotong University. Dr. Li received his B.Eng. degree in electrical and electronic engineering from Beijing Jiaotong University in 2003, and M.Sci. and Ph.D. degrees in computer science from the University of Adelaide, in 2006 and 2010, respectively. Dr. Li's research interests include big data analysis, privacy preserving and information security and intelligent transportation. Dr. Li has published more than 100 research papers in various journals and refereed conferences. He has organized several international conferences and workshops and has also served as a program committee member for several major international conferences.

Hairong Dong (M'02-SM'12) received the Ph.D. degree from Peking University in 2002. She is currently the deputy director of the National Engineering Research Center for Rail Transportation Operation Control System, and also a professor with the State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, China. Her research interests include intelligent transportation systems, automatic train operation, intelligent dispatching and complex network applications. Prof. Dong was a visiting scholar with the University of Southampton in 2006 and the University of Hong Kong in 2008. She was also a visiting professor with the KTH Royal Institute of Technology in 2011.
Prof. Dong is currently the Fellow of the Chinese Automation Congress and the Co-Chair of the Technical Committee on Railroad Systems and Applications of the IEEE Intelligent Transportation Systems Society. She serves as Associate Editor for IEEE Transactions on Intelligent Transportation Systems, IEEE Intelligent Transportation Systems Magazine, and Journal of Intelligent & Robotic Systems.

Haifeng Song (M'19) received his B.S. and master degrees from Beijing Jiaotong University, China, in 2011 and 2014, respectively. He received the Ph.D. degree from Technische Universitat Braunschweig, Germany, in 2018. He has joined the Institute for Traffic Safety and Automation Engineering, Technische Universitat Braunschweig, Germany, as a visiting scientist since 2014. He is an Associate Professor with the School of Electronic and Information Engineering, Beijing Jiaotong University, China.
He specializes in safety and security of transportation systems, his current research interests include railway control system, formal method, intelligent control, and transportation modeling. He has been involved in several national and international research projects dealing with system safety and system evaluation. He is a member of IEEE Intelligent Transportation Systems Society, Chinese Association of Automation, China Institute of Communications and a reviewer for international journals.