

Deep Learning on Traffic Prediction: Methods, Analysis and Future Directions

Xueyan Yin, Genze Wu, Jinze Wei, Yanming Shen^{ID}, Heng Qi^{ID}, *Member, IEEE*, and Baocai Yin, *Member, IEEE*

Abstract—Traffic prediction plays an essential role in intelligent transportation system. Accurate traffic prediction can assist route planing, guide vehicle dispatching, and mitigate traffic congestion. This problem is challenging due to the complicated and dynamic spatio-temporal dependencies between different regions in the road network. Recently, a significant amount of research efforts have been devoted to this area, especially deep learning method, greatly advancing traffic prediction abilities. The purpose of this paper is to provide a comprehensive survey on deep learning-based approaches in traffic prediction from multiple perspectives. Specifically, we first summarize the existing traffic prediction methods, and give a taxonomy. Second, we list the state-of-the-art approaches in different traffic prediction applications. Third, we comprehensively collect and organize widely used public datasets in the existing literature to facilitate other researchers. Furthermore, we give an evaluation and analysis by conducting extensive experiments to compare the performance of different methods on a real-world public dataset. Finally, we discuss open challenges in this field.

Index Terms—Traffic prediction, deep learning, spatial-temporal dependency modeling.

I. INTRODUCTION

THE modern city is gradually developing into a smart city. The acceleration of urbanization and the rapid growth of urban population bring great pressure to urban traffic management. Intelligent Transportation System (ITS) is an indispensable part of smart city, and traffic prediction is an important component of ITS. Accurate traffic prediction is essential to many real-world applications. For example, traffic flow prediction can help city alleviate congestion; car-hailing demand prediction can prompt car-sharing companies pre-allocate cars to high demand regions. The growing available traffic related datasets provide us potential new perspectives to explore this problem.

Manuscript received April 29, 2020; revised November 12, 2020; accepted January 15, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant U1811463 and Grant 61772112, and in part by the Innovation Foundation of Science and Technology of Dalian under Grant 2018J11CY010 and Grant 2019J12GX037. The Associate Editor for this article was Y. Kamarianakis. (*Corresponding author: Yanming Shen.*)

Xueyan Yin, Genze Wu, Jinze Wei, and Heng Qi are with the School of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, China.

Yanming Shen is with the School of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, China, and also with the Key Laboratory of Intelligent Control and Optimization for Industrial Equipment, Ministry of Education, Dalian University of Technology, Dalian 116024, China (e-mail: shen@dlut.edu.cn).

Baocai Yin is with the School of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, China, and also with the Peng Cheng Laboratory, Shenzhen 518055, China.

Digital Object Identifier 10.1109/TITS.2021.3054840

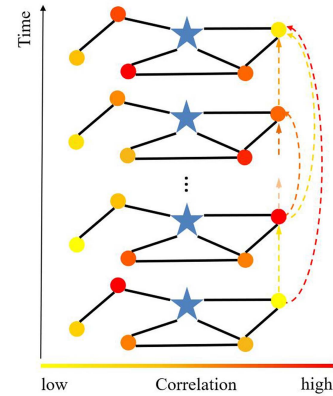


Fig. 1. Complex spatio-temporal correlations. The nodes represent different locations in the road network, and the blue star node represents the predicted target. The darker the color, the greater the spatial correlation with the target node. The dotted line shows the temporal correlation between different time steps.

A. Challenges

Traffic prediction is very challenging, mainly affected by the following complex factors:

(1) Because traffic data is spatio-temporal, it is constantly changing with time and space, and has complex and dynamic spatio-temporal dependencies.

- **Complex spatial dependencies.** Fig.1 demonstrates that the influence of different positions on the predicted position is different, and the influence of the same position on the predicted position is also varying with time. The spatial correlation between different positions is highly dynamic.
- **Dynamic temporal dependencies.** The observed values at different times of the same position show non-linear changes, and the traffic state of the far time step sometimes has greater influence on the predicted time step than that of the recent time step, as shown in Fig.1. Meanwhile, [1] pointed out that traffic data usually presents periodicity, such as closeness, period and trend. Therefore, how to select the most relevant historical observations for prediction remains a challenging problem.

(2) **External factors.** Traffic spatio-temporal sequence data is also influenced by some external factors, such as weather conditions, events or road attributes.

Since traffic data shows strong dynamic correlation in both spatial and temporal dimensions, it is an important research topic to mine the non-linear and complicated spatial-temporal

patterns, making accurate traffic predictions. Traffic prediction involves various application tasks. Here, we list the main application tasks of the existing traffic prediction work, which are as follows:

- Flow
Traffic flow refers to the number of vehicles passing through a given point on the roadway in a certain period of time.
- Speed
The actual speed of vehicles is defined as the distance it travels per unit of time. Most of the time, due to factors such as geographical location, traffic conditions, driving time, environment and personal circumstances of the driver, each vehicle on the roadway will have a speed that is somewhat different from those around it.
- Demand
The problem is how to use historical requesting data to predict the number of requests for a region in a future time step, where the number of start/pick-up or end/drop-off is used as a representation of the demand in a region at a given time.
- Travel time
In the case of obtaining the route of any two points in the road network, estimating the travel time is required. In general, the travel time should include the waiting time at the intersection.
- Occupancy
The occupancy rate explains the extent to which vehicles occupy road space, and is an important indicator to measure whether roads are fully utilized.

B. Related Surveys on Traffic Prediction

There are a few recent surveys that have reviewed the literatures on traffic prediction in certain contexts from different perspectives. [2] reviewed the methods and applications from 2004 to 2013, and discussed ten challenges that were significant at the time. It is more focused on considering short-term traffic prediction and the literatures involved are mainly based on the traditional methods. Another work [3] also paid attention to short-term traffic prediction, which briefly introduced the techniques used in traffic prediction and gave some research suggestions. [4] provided sources of traffic data acquisition, and mainly focused on traditional machine learning methods. [5] outlined the significance and research directions of traffic prediction. [6] and [7] summarized relevant models based on classical methods and some early deep learning methods. Alexander *et al.* [8] presented a survey of deep neural network for traffic prediction. It discussed three common deep neural architectures, including convolutional neural network, recurrent neural network, and feedforward neural network. However, some recent advancements, e.g., graph-based deep learning, were not covered in [8]. [9] is an overview of graph-based deep learning architecture, with applications in the general traffic domain. [10] provided a survey focusing specifically on the use of deep learning models for analyzing traffic data. However, it only investigates the traffic flow prediction. In general, different traffic prediction

tasks have common characteristics, and it is beneficial to consider them jointly. Therefore, there is still a lack of broad and systematic survey on exploring traffic prediction in general.

C. Our Contributions

To our knowledge, this is the first comprehensive survey on deep learning-based works in traffic prediction from multiple perspectives, including approaches, applications, datasets, experiments, analysis and future directions. Specifically, the contributions of this survey can be summarized as follows:

- We first do a taxonomy for existing approaches, describing their key design choices.
- We collect and summarize available traffic prediction datasets, which provide a useful pointer for other researches.
- We perform a comparative experimental study to evaluate different models, identifying the most effective component.
- We further discuss possible limitations of current solutions, and list promising future research directions.

D. A Taxonomy of Existing Approaches

After years of efforts, the research on traffic prediction has achieved great progresses. In light of the development process, these methods can be broadly divided into two categories: classical methods and deep learning-based methods. Classical methods include statistical methods and traditional machine learning methods. The statistical method is to build a data-driven statistical model for prediction. The most representative algorithms are Historical Average (HA), Auto-Regressive Integrated Moving Average (ARIMA) [11], and Vector Auto-Regressive (VAR) [12]. Nevertheless, these methods require data to satisfy certain assumptions, and time-varying traffic data is too complex to satisfy these assumptions. Moreover, these methods are only applicable to relatively small datasets. Later, a number of traditional machine learning methods, such as Support Vector Regression (SVR) [13] and Random Forest Regression (RFR) [14], were proposed for traffic prediction problem. Such methods have the ability to process high-dimensional data and capture complex non-linear relationships.

It was not until the advent of deep learning-based methods that the full potential of artificial intelligence in traffic prediction was developed [15]. This technology studies how to learn a hierarchical model to map the original input directly to the expected output [16]. In general, deep learning models stack up basic learnable blocks or layers to form a deep architecture, and the entire network is trained end-to-end. Several architectures have been developed to handle large-scale and complex spatio-temporal data. Generally, Convolutional Neural Network (CNN) [17] is employed to extract spatial correlation of the grid-structured data described by images or videos, and Graph Convolutional Network (GCN) [18] extends convolution operation to more general graph-structured data, which is more suitable to represent the traffic network structure. Furthermore, Recurrent Neural Network (RNN) [19], [20]

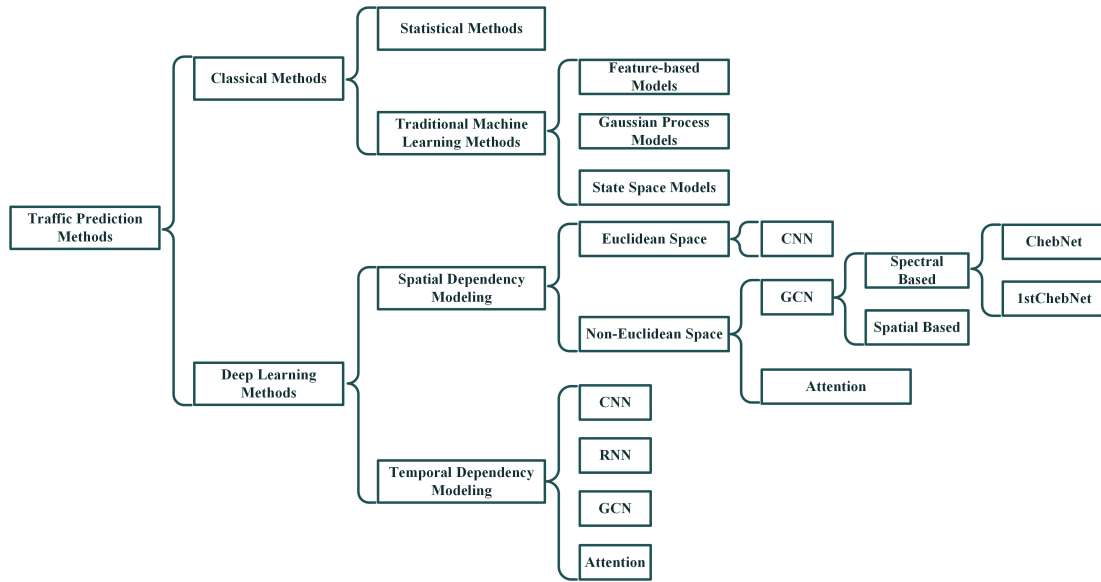


Fig. 2. Key techniques of traffic prediction methods.

and its variants LSTM [21] or GRU [22] are commonly utilized to model temporal dependency. Here, we summarize the key techniques commonly used in existing traffic prediction methods, as shown in Fig. 2.

E. Organization of This Survey

The rest of this paper is organized as follows. Section II covers the classical methods for traffic prediction. Section III reviews the work based on deep learning methods for traffic prediction, including the commonly used methods of modeling spatial correlation and temporal correlation, as well as some other new variants. Section IV lists some representative results in each task. Section V collects and organizes related datasets and commonly used external data types for traffic prediction. Section VI provides some comparisons and evaluates the performance of the relevant methods. Section VII discusses several significant and important directions of future traffic prediction. Finally, we conclude this paper in Section VIII.

II. CLASSICAL METHODS

Statistical and traditional machine learning models are two major representative data-driven methods for traffic prediction. In time-series analysis, autoregressive integrated moving average (ARIMA) [11] and its variants are one of the most consolidated approaches based on classical statistics and have been widely applied for traffic prediction problems ([11], [23]–[27]). However, these methods are generally designed for small datasets, and are not suitable to deal with complex and dynamic time series data. In addition, since usually only temporal information is considered, the spatial dependency of traffic data is ignored or barely considered.

Traditional machine learning methods, which can model more complex data, are broadly divided into three categories: feature-based models, Gaussian process models and state space models. Feature-based methods solve traffic prediction

problem ([28]–[30]) by training a regression model based on human-engineered traffic features. These methods are simple to implement and can provide predictions in some practical situations. Gaussian process models the inner characteristics of traffic data through different kernel functions, which need to contain spatial and temporal correlations simultaneously. Although this kind of methods is proved to be effective and feasible in traffic prediction ([31]–[33]), compared to feature-based models, they generally have higher computational load and storage pressure, which is not appropriate when a mass of training samples are available. State space models assume that the observations are generated by Markovian hidden states. The advantage of this model is that it can naturally model the uncertainty of the system and better capture the latent structure of the spatio-temporal data. However, the overall non-linearity of these models ([34]–[48]) is limited, and most of the time they are not optimal for modeling complex and dynamic traffic data. Table I summarizes some recent representative classical approaches.

III. DEEP LEARNING METHODS

Deep learning models exploit much more features and complex architectures than the classical methods, and can achieve better performance. In Table II, we summarize the deep learning architectures in the existing traffic prediction literature, and we will review these commonly components in this section.

A. Modeling Spatial Dependency

CNN. A series of studies have applied CNN to capture spatial correlations in traffic networks from two-dimensional spatio-temporal traffic data [3]. Since the traffic network is difficult to be described by 2D matrices, several researches try to convert the traffic network structure at different times into images and divide these images into standard grids, with

TABLE I
CLASSICAL METHODS

Category		Application task	Approach
Statistical methods		Flow Demand	[11], [23], [26], [27] [24], [25]
Traditional machine learning methods	Feature-based models	Flow Demand	[30] [28], [29]
		Flow Speed Demand Occupancy	[31] [33] [32] [32]
	Gaussian process models	Flow Speed Demand Occupancy	[31] [33] [32] [32]
		Flow Speed Demand Travel time Occupancy	[34], [35], [38]–[40], [45]–[48] [36], [42], [43] [37] [44] [41], [44]
	State space models	Flow Speed Demand Travel time Occupancy	[34], [35], [38]–[40], [45]–[48] [36], [42], [43] [37] [44] [41], [44]

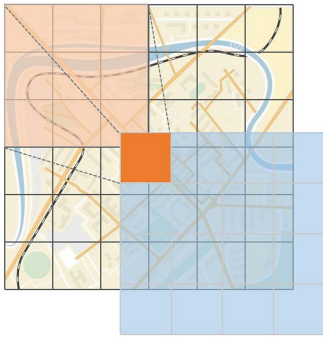


Fig. 3. 2D Convolution. Each grid in the image is treated as a region, where neighbors are determined by the filter size. The 2D convolution operates between a certain region and its neighbors. The neighbors of a region are ordered and have a fixed size.

each grid representing a region. In this way, CNNs can be used to learn spatial features among different regions.

As shown in Fig. 3, each region is directly connected to its nearby regions. With a 3×3 window, the neighborhood of each region is its surrounding eight regions. The positions of these eight regions indicate an ordering of a region's neighbors. A filter is then applied to this 3×3 patch by taking the weighted average of the central region and its neighbors across each channel. Due to the specific ordering of neighboring regions, the trainable weights are able to be shared across different locations.

In the division of traffic road network structure, there are many different definitions of positions according to different granularity and semantic meanings. [1] divided a city into $I \times J$ grid maps based on the longitude and latitude where a grid represented a region. Then, a CNN was applied to extract the spatial correlation between different regions for traffic flow prediction.

GCN. Traditional CNN is limited to modeling Euclidean data, and GCN is therefore used to model non-Euclidean spatial structure data, which is more in line with the structure of traffic road network. GCN generally consists of two type of methods, spectral-based and spatial-based methods. Spectral-based approaches define graph convolutions by introducing filters from the perspective of graph signal processing where the graph convolution operation is interpreted as removing noise from graph signals. Spatial-based approaches formulate

graph convolutions as aggregating feature information from neighbors. In the following, we will introduce spectral-based GCNs and spatial-based GCNs respectively.

1) *Spectral Methods:* Bruna *et al.* [18] first developed spectral network, which performed convolution operation for graph data from spectral domain by computing the eigen-decomposition of the graph Laplacian matrix \mathbf{L} . Specifically, the graph convolution operation $*_G$ of a signal \mathbf{x} with a filter $\mathbf{g} \in \mathbb{R}^N$ can be defined as:

$$\mathbf{x} *_G \mathbf{g} = \mathbf{U} \left(\mathbf{U}^T \mathbf{x} \odot \mathbf{U}^T \mathbf{g} \right), \quad (1)$$

where \mathbf{U} is the matrix of eigenvectors of normalized graph Laplacian \mathbf{L} , which is defined as $\mathbf{L} = \mathbf{I}_N - \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^T$, \mathbf{D} is the diagonal matrix, $\mathbf{D}_{ii} = \sum_j (\mathbf{A}_{ij})$, \mathbf{A} is the adjacency matrix of the graph, $\mathbf{\Lambda}$ is the diagonal matrix of eigenvalues, $\mathbf{\Lambda} = \lambda_i$. If we denote a filter as $\mathbf{g}_\theta = \text{diag}(\mathbf{U}^T \mathbf{g})$ parameterized by $\theta \in \mathbb{R}^N$, the graph convolution can be simplified as:

$$\mathbf{x} *_G \mathbf{g} = \mathbf{U} \mathbf{g}_\theta \mathbf{U}^T \mathbf{x}, \quad (2)$$

where a graph signal \mathbf{x} is filtered by \mathbf{g} with multiplication between \mathbf{g} and graph transform $\mathbf{U}^T \mathbf{x}$. Though the computation of filter \mathbf{g} in graph convolution can be expensive due to $\mathcal{O}(n^2)$ multiplications with matrix \mathbf{U} , two approximation strategies have been successively proposed to solve this issue.

ChebNet. Defferrard *et al.* [49] introduced a filter as Chebyshev polynomials of the diagonal matrix of eigenvalues, i.e., $\mathbf{g}_\theta = \sum_{i=1}^K \theta_i \mathbf{T}_i(\tilde{\Lambda})$, where $\theta \in \mathbb{R}^K$ is now a vector of Chebyshev coefficients, $\tilde{\Lambda} = \frac{2}{\lambda_{\max}} \mathbf{\Lambda} - \mathbf{I}_N$, and λ_{\max} denotes the largest eigenvalue. The Chebyshev polynomials are defined as $\mathbf{T}_k(\mathbf{x}) = 2\mathbf{x}\mathbf{T}_{k-1}(\mathbf{x}) - \mathbf{T}_{k-2}(\mathbf{x})$ with $\mathbf{T}_0\mathbf{x} = 1$ and $\mathbf{T}_1(\mathbf{x}) = \mathbf{x}$. Then, the convolution operation of a graph signal \mathbf{x} with the defined filter \mathbf{g}_θ is:

$$\begin{aligned} \mathbf{x} *_G \mathbf{g}_\theta &= \mathbf{U} \left(\sum_{i=1}^K \theta_i \mathbf{T}_i(\tilde{\Lambda}) \right) \mathbf{U}^T \mathbf{x} \\ &= \sum_{i=1}^K \theta_i \mathbf{T}_i(\tilde{\mathbf{L}}) \mathbf{x}, \end{aligned} \quad (3)$$

where $\tilde{\mathbf{L}} = \frac{2}{\lambda_{\max}} \mathbf{L} - \mathbf{I}_N$.

First order of ChebNet (1stChebNet). An first-order approximation of ChebNet introduced by Kipf and Welling [50] further simplified the filtering by assuming $K = 1$ and $\lambda_{max} = 2$, we can obtain the following simplified expression:

$$\mathbf{x} *_G \mathbf{g}_\theta = \theta_0 \mathbf{x} - \theta_1 \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \mathbf{x}, \quad (4)$$

where θ_0 and θ_1 are learnable parameters. After further assuming these two free parameters with $\theta = \theta_0 = -\theta_1$. This can be obtained equivalently in the following matrix form:

$$\mathbf{x} *_G \mathbf{g}_\theta = \theta \left(\mathbf{I}_N + \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \right) \mathbf{x}. \quad (5)$$

To avoid numerical instabilities and exploding/vanishing gradients due to stack operations, another normalization technique is introduced: $\mathbf{I}_N + \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \rightarrow \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}}$, with $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_N$ and $\tilde{\mathbf{D}}_{ii} = \sum_j \tilde{\mathbf{A}}_{ij}$. Finally, a graph convolution operation can be changed to:

$$\mathbf{Z} = \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{X} \Theta, \quad (6)$$

where $\mathbf{X} \in \mathbb{R}^{N \times C}$ is a signal, $\Theta \in \mathbb{R}^{C \times F}$ is a matrix of filter parameters, C is the input channels, F is the number of filters, and Z is the transformed signal matrix.

To fully utilize spatial information, [51] modeled the traffic network as a general graph rather than treating it as grids, where the monitoring stations in a traffic network represent the nodes in the graph, the connections between stations represent the edges, and the adjacency matrix is computed based on the distances among stations, which is a natural and reasonable way to formulate the road network. Afterwards, two graph convolution approximation strategies based on spectral methods were used to extract patterns and features in the spatial domain, and the computational complexity was also reduced. [52] first used graphs to encode different kinds of correlations among regions, including neighborhood, functional similarity, and transportation connectivity. Then, three groups of GCN based on ChebNet were used to model spatial correlations respectively, and traffic demand prediction was made after further integrating temporal information.

2) *Spatial Methods:* Spatial methods define convolutions directly on the graph through the aggregation process that operates on the central node and its neighbors to obtain a new representation of the central node, as depicted by Fig.4. In [53], traffic network was firstly modeled as a directed graph, the dynamics of the traffic flow was captured based on the diffusion process. Then a diffusion convolution operation is applied to model the spatial correlation, which is a more intuitive interpretation and proves to be effective in spatial-temporal modeling. Specifically, diffusion convolution models the bidirectional diffusion process, enabling the model to capture the influence of upstream and downstream traffic. This process can be defined as:

$$\mathbf{X}_{:,p} *_G f_\theta = \sum_{k=0}^{K-1} \left(\theta_{k1} \left(\mathbf{D}_O^{-1} \mathbf{A} \right)^k + \theta_{k2} \left(\mathbf{D}_I^{-1} \mathbf{A}^T \right)^k \right) \mathbf{X}_{:,p}, \quad (7)$$

where $\mathbf{X} \in \mathbb{R}^{N \times P}$ is the input, P represents the number of input features of each node. $*_G$ denotes the diffusion convolution, k is the diffusion step, f_θ is a filter and $\theta \in \mathbb{R}^{K \times 2}$

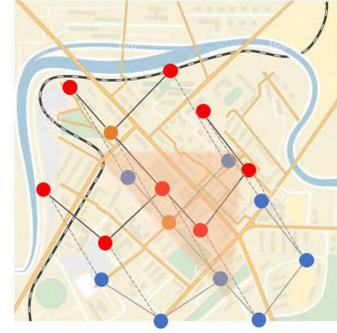


Fig. 4. Spatial-based graph convolution network. Each node in the graph can represent a region in the traffic network. To get a hidden representation of a certain node (e.g. the orange node), GCN aggregates feature information from its neighbors (shaded area). Unlike grid data in 2D images, the neighbors of a region are unordered and varies in size.

are learnable parameters. \mathbf{D}_O and \mathbf{D}_I are out-degree and in-degree matrices respectively. To allow multiple input and output channels, DCRNN [53] proposes a diffusion convolution layer, defined as:

$$\mathbf{Z}_{:,p} = \sigma \left(\sum_{p=1}^P \mathbf{X}_{:,p} *_G f_{\Theta_{q,p,:}} \right), \quad (8)$$

where $\mathbf{Z} \in \mathbb{R}^{N \times Q}$ is the output, $\Theta \in \mathbb{R}^{Q \times P \times K \times 2}$ parameterizes the convolutional filter, Q is the number of output features, σ is the activation function. Based on the diffusion convolution process, [54] designed a new neural network layer that can map the transformation of different dimensional features and extract patterns and features in spatial domain. [55] modified the diffusion process in [53] by utilizing a self-adaptive adjacency matrix, which allowed the model to mine hidden spatial dependency by itself. [56] introduced the notion of aggregation to define graph convolution. This operation can assemble the features of each node with its neighbors. The aggregate function is a linear combination whose weights are equal to the weights of the edges between the node and its neighbors. This graph convolutional operation can be expressed as follow:

$$h^{(l)} = \sigma(\mathbf{A} h^{(l-1)} \mathbf{W} + b), \quad (9)$$

where $h^{(l-1)}$ is the input of the l -th graph convolutional layer, \mathbf{W} and b are parameters, and σ is the activation function.

Attention. Attention mechanism is first proposed for natural language processing [57], and has been widely used in various fields. The traffic condition of a road is affected by other roads with different impacts. Such impact is highly dynamic, changing over time. To model these properties, the spatial attention mechanism is often used to adaptively capture the correlations between regions in the road network ([58]–[66]). The key idea is to dynamically assign different weights to different regions at different time steps. For the sake of simplicity, we ignore time coordinates for the moment. Attention mechanism operates on a set of input sequence $x = (x_1, \dots, x_n)$ with n elements where $x_i \in \mathbb{R}^{d_x}$, and computes a new sequence $z = (z_1, \dots, z_n)$ with the same

length where $z_i \in \mathbb{R}^{d_z}$. Each output element z_i is computed as a weighted sum of a linear transformed input elements:

$$z_i = \sum_{j=1}^n a_{ij} x_j. \quad (10)$$

The weight coefficient a_{ij} indicates the importance of x_i to x_j , and it is computed by a softmax function:

$$a_{ij} = \frac{\exp e_{ij}}{\sum_{k=1}^n \exp e_{ik}}, \quad (11)$$

where e_{ij} is computed using a compatibility function that compares two input elements:

$$e_{ij} = v^\top \tanh(x_i W^Q + x_j W^k + b), \quad (12)$$

and generally Perceptron is chosen for the compatibility function. Here, the learnable parameters are v , W^Q , W^k and b . This mechanism has proven effective, but when the number of elements n in a sequence is large, we need to calculate n^2 weight coefficients, and therefore the time and memory consumption are heavy.

In traffic speed prediction, [60] used attention mechanism to dynamically capture the spatial correlation between the target region and the first-order neighboring regions of the road network. [67] combined the GCN based on ChebNet with attention mechanism to make full use of the topological properties of the traffic network and dynamically adjust the correlations between different regions.

B. Modeling Temporal Dependency

CNN. [68] first introduced the fully convolutional model for sequence to sequence learning. A representative work in traffic research, [51] applied purely convolutional structures to simultaneously extract spatio-temporal features from graph-structured time series data. In addition, dilated causal convolution is a special kind of standard one-dimensional convolution. It adjusts the size of the receptive field by changing the value of the dilation rate, which is conducive to capture the long-term periodic dependence. [69] and [70] therefore adopted the dilated causal convolution as the temporal convolution layer of their models to capture a node's temporal trends. Compared to recurrent models, convolutions create representations for fixed size contexts, however, the effective context size of the network can easily be made larger by stacking several layers on top of each other. This allows to precisely control the maximum length of dependencies to be modeled. The convolutional network does not rely on the calculation of the previous time step, so it allows parallelization of every element in the sequence, which can make better use of GPU hardware, and easier to optimize. This is superior to RNNs, which maintain the entire hidden state of the past, preventing parallel calculations in a sequence.

RNN. RNN and its variant LSTM or GRU, are neural networks for processing sequential data. To model the non-linear temporal dependency of traffic data, RNN-based approaches have been applied to traffic prediction [3]. These models rely on the order of data to process data in turn, and therefore one disadvantage of these models is that when modeling long

sequences, their ability to remember what they learned before many time steps may decline.

In RNN-based sequence learning, a special network structure known as encoder-decoder has been applied for traffic prediction ([53], [58], [61]–[66], [71]–[79]). The key idea is to encode the source sequence as a fixed-length vector and use the decoder to generate the prediction.

$$\mathbf{s} = f(\mathcal{F}_t; \theta_1), \quad (13)$$

$$\hat{\mathbf{X}}_{t+1:t+L} = g(\mathbf{s}; \theta_2), \quad (14)$$

where f is the encoder and g is the decoder. \mathcal{F}_t denotes the input information available at timestamp t , \mathbf{s} is a transformed semantic vector representation, $\hat{\mathbf{X}}_{t+1:t+L}$ is the value of L -step-ahead prediction, θ_1 and θ_2 are learning parameters.

One potential problem with encoder-decoder structure is that regardless of the length of the input and output sequences, the length of semantic vector \mathbf{s} between encoding and decoding is always fixed, and therefore when the input information is too long, some information will be lost.

Attention. To resolve the above issue, an important extension is to use an attention mechanism on time axis, which can adaptively select the relevant hidden states of the encoder to produce output sequence. This is similar to attention in the spatial methods. Such a temporal attention mechanism can not only model the non-linear correlation between the current traffic condition and the previous observations at a certain position in the road network, but also model the long-term sequence data to solve the deficiencies of RNN.

[62] designed a temporal attention mechanism to adaptively model the non-linear correlations between different time steps. [67] incorporated a standard convolution and attention mechanism to update the information of a node by fusing the information at the neighboring time steps, and semantically express the dependency intensity between different time steps. Considering that traffic data is highly periodic, but not strictly periodic, [80] designed a periodically shifted attention mechanism to deal with long-term periodic dependency and periodic temporal shifting.

GCN. Song *et al.* first constructed a localized spatio-temporal graph that includes both temporal and spatial attributes, and then used the proposed spatial-based GCN method to model the spatio-temporal correlations simultaneously [56].

C. Joint Spatio-Temporal Relationships Modeling

As shown in Table II, most methods use a hybrid deep learning framework, which combines different types of techniques to capture the spatial dependencies and temporal correlations of traffic data separately. They assume that the relations of geographic information and temporal information are independent and do not consider their joint relations. Therefore, the spatial and temporal correlations are not fully exploited to obtain better accuracy. To solve this limitation, researchers have attempted to integrate spatial and temporal information into an adjacency graph matrix or tensor. For example, [56] got a localized spatio-temporal graph by connecting all nodes with themselves at the previous moment and the next moment.

TABLE II
CATEGORIZATION FOR THE COVERED DEEP LEARNING LITERATURE

Application task	Spatial modeling type	Temporal modeling type	Approach
Flow	CNN	–	[1], [86]–[90]
		RNN	[70], [74], [91]–[94]
	1stChebNet	RNN	[79], [95]
	ChebNet	CNN (Causal CNN)	[69]
		RNN	[96]
	GCN+Attention	CNN (1-D Conv) +Attention	[67]
	Attention only	–	[97]
		RNN	[58]
		RNN+Attention	[65], [66]
		Attention only	[62]
Speed	–	RNN	[98], [99]
	CNN	RNN	[100], [101]
	1stChebNet	CNN (1-D Conv)	[51]
		RNN	[102]
	ChebNet	CNN (1-D Conv)	[51]
		(2-D Conv)	[103]
		RNN	[73], [96], [104], [105]
		RNN+Attention	[75]
	GCN(spatial-based)	CNN (Causal CNN)	[55]
		RNN	[53], [54]
	GCN+Attention	CNN (1-D Conv)	[106]
		RNN	[107]
	Attention only	RNN	[58], [60]
		Attention only	[62], [63]
Demand	–	RNN	[108]
	CNN	–	[109]
		RNN	[72], [110]–[112]
		RNN+Attention	[80]
	1stChebNet	Attention only	[77]
		RNN	[76], [113]
	ChebNet	RNN	[52]
	Attention only	RNN	[59]
		Attention only	[61], [64]
Travel time	–	RNN	[114], [115]
	CNN	RNN	[116]
	ChebNet	CNN (1-D Conv)	[117]
Occupancy	–	RNN	[78]
	CNN	RNN	[118]

TABLE III
CATEGORIZATION FOR THE COVERED DEEP LEARNING PLUS CLASSICAL
LITERATURE

Application task	Approach	Spatio-temporal modeling
Flow	[85]	State space model+MLP
	[83]	State space model+CNN+RNN
Demand	[83]	State space model+CNN+RNN
Occupancy	[82]	State space model+RNN
	[84]	State space model+CNN

According to the topological structure of the localized spatial-temporal graph, the correlations between each node and its spatio-temporal neighbors can be captured directly. In [81], Fang *et al.* constructed three matrices for the historical traffic conditions of different links, the features of the neighbor links, and the features of the historical time slots, in which each row of the matrix corresponds to the information of a link. Finally, these three matrices were concatenated into a matrix and reshaped into a 3D spatio-temporal tensor. Attention mechanism was then used to obtain the relations between the traffic conditions.

D. Deep Learning Plus Classical Models

Recently, more and more researches are combining deep learning with classical methods, and some advanced methods have been used in traffic prediction ([82]–[85]). This kind of method not only makes up for the weak ability of non-linear representation of classical models but also makes up for the poor interpretability of deep learning methods. [82] proposed a method based on the generation model of state space and the inference model based on filtering, using deep neural networks to realize the non-linearity of the emission and the transition models, and using the recurrent neural network to realize the dependence over time. Such a non-linear network based parameterization provides the flexibility to deal with arbitrary data distribution. [83] proposed a deep learning framework that introduced matrix factorization method into deep learning model, which can model the latent region functions along with the correlations among regions, and further improve the model capability of the citywide flow prediction. [84] developed a hybrid model that associated a global matrix decomposition model regularized by a temporal deep network with a local deep temporal model that captured patterns specific to each dimension. Global and local models are combined through a

data-driven attention mechanism for each dimension. Therefore, global patterns of the data can be utilized and combined with local calibration for better prediction. [85] combined a latent model and multi-layer perceptrons (MLP) to design a network for addressing multivariate spatio-temporal time series prediction problems. The model captures the dynamics and correlations of multiple series at the spatial and temporal levels. Table III summarizes relevant literatures in terms of deep learning plus classical methods.

E. Limitations of the Deep Learning-Based Method

The strengths of the deep neural network model make it very attractive and indeed greatly promote the progress in the field of traffic prediction. However, it also possess several disadvantages compared with classical methods.

- High data demand. Deep learning is highly data-dependent, and typically the larger the amount of data, the better it performs. In many cases, such data is not readily available, for example, some cities may release taxi data for multiple years, while others release data for just a few days.
- High computational complexity. Deep learning requires high computing power, and ordinary CPUs can no longer meet the requirements of deep learning. The mainstream computing uses GPU and TPU. At the same time, with the increase of model complexity and the number of parameters, the demand for memory is also gradually increasing. In general, deep neural networks are more computationally expensive than classical algorithms.
- Lack of interpretability. Deep learning models are mostly considered as “black-boxes” that lack interpretability. In general, the prediction accuracy of deep learning models is higher than that of classical methods. However, there is no explanation as to why these results are obtained or how parameters can be determined to make the results better.

IV. REPRESENTATIVE RESULTS

In this section, we summarize some representative results of different application tasks. Based on the literature studied on different tasks, we list the current best performance methods under commonly used public datasets, as shown in Table IV. We can have the following observations: First, the results on different datasets vary greatly under the same prediction task. For example, in the demand prediction task, the NYC Taxi and TaxiBJ datasets obtained the accuracy of 8.385 and 17.24, respectively, under the same time interval and prediction time. Under the same condition of the prediction task and the dataset, the performance decreases with the increase of prediction time, as shown in the speed prediction results on Q-Traffic. For the dataset of the same data source, due to the different time and region selected, it also has a greater impact on the accuracy, e.g., related datasets based on PeMS under the speed prediction task. Second, in different prediction tasks, the accuracy of speed prediction task can reach above 90% in general, which is significantly higher than other tasks whose

accuracy rate is close to or more than 80%. Therefore, there is still much room for improvement in these tasks.

Some companies are currently conducting intelligent transportation research, such as amap, DiDi, and Baidu maps. According to amap technology annual in 2019 [119], amap has carried out the exploration and practice of deep learning in the prediction of the historical speed of amap driving navigation, which is different from the common historical average method and takes into account the timeliness and annual periodicity characteristics presented in the historical data. By introducing the Temporal Convolutional Network (TCN) [120] model for industrial practice, and combining feature engineering (extracting dynamic and static features, introducing annual periodicity, etc.), the shortcomings of existing models are successfully solved. The arrival time of a given week is measured based on the order data, and it has a badcase rate of 10.1%, which is 0.9% lower than the baseline. For the travel time prediction in the next hour, [117] designed a multi-model architecture to infer the future travel time by adding contextual information using the upcoming traffic flow data. Using anonymous user data from amap, MAPE can be reduced to around 16% in Beijing.

The Estimated Time of Arrival (ETA), supply and demand and speed prediction are the key technologies in DiDi's platform. DiDi has applied artificial intelligence technology in ETA, reduced MAPE index to 11% by utilizing neural network and DiDi's massive order data, and realized the ability to provide users with accurate expectation of arrival time and multi-strategy path planning under real-time large-scale requests. In the prediction and scheduling, DiDi has used deep learning model to predict the difference between supply and demand after some time in the future, and provided driver scheduling service. The prediction accuracy of the gap between supply and demand in the next 30 minutes has reached 85%. In the urban road speed prediction task, DiDi proposed a prediction model based on driving trajectory calibration [121]. Through comparison experiments based on Chengdu and Xi'an data in the DiDi gaia dataset, it was concluded that the overall MSE indicator for speed prediction was reduced to 3.8 and 3.4.

Baidu has solved the traffic prediction task of online route queries by integrating auxiliary information into deep learning technology, and released a large-scale traffic prediction dataset from Baidu Map with offline and online auxiliary information [73]. The overall MAPE and 2-hour MAPE of speed prediction on this dataset decreased to 8.63% and 9.78%, respectively. In [81], the researchers proposed an end-to-end neural framework as an industrial solution for the travel time prediction function in mobile map applications, aiming at exploration of spatio-temporal relation and contextual information in traffic prediction. The MAPE in Taiyuan, Hefei and Huizhou, sampled on the Baidu maps, can be reduced to 21.79%, 25.99% and 27.10% respectively, which proves the superiority of the model. The model is already in production on Baidu maps and successfully handles tens of billions of requests a day.

V. PUBLIC DATASETS

High-quality datasets are essential for accurate traffic forecasting. In this section, we comprehensively summarize the

TABLE IV
PREDICTION PERFORMANCE STATISTICS FOR DIFFERENT TASKS

Application task	Dataset	Time interval	Prediction window	MAPE	RMSE
Flow	TaxiBJ	30min	30min	25.97% [90]	15.88 [90]
	PeMSD3	5min	60min	16.78% [56]	29.21 [56]
	PeMSD4	5min	60min	11.09% [65]	31.00 [65]
	PEMS07	5min	60min	10.21% [56]	38.58 [53]
	PeMSD8	5min	60min	8.31% [65]	24.74 [65]
	NYC Bike	60min	60min	–	6.33 [87]
Speed	T-Drive	60min	60/120/180min	–	29.9/34.7/37.1 [58]
	METR-LA	5min	5/15/30/60min	4.90% [54]/6.80%/8.30%/10.00% [107]	3.57 [54] 5.12/6.17/7.30 [107]
	PeMS-BAY	5min	15/30/60min	2.73% [55]/3.63% [62]/4.31% [62]	2.74 [55]/3.70 [55]/4.32 [62]
	PeMSD4	5min	15/30/45/60min	2.68% [53]/3.71% [53]/4.42%/4.85% [106]	2.93/3.92/4.47/4.83 [106]
	PeMSD7	5min	15/30/45/60min	5.14%/7.18%/8.51%/9.60% [106]	3.98/5.47/6.39/7.09 [106]
	PeMSD7(M)	5min	15/30/45min	5.24%/7.33%/8.69% [51]	4.04/5.70/6.77 [51]
	PeMSD8	5min	15/30/45/60min	2.24%/3.02%/3.51%/3.89% [106]	2.45/3.28/3.75/4.11 [106]
	SZ-taxi	15min	15/30/45/60min	–	3.92/3.96/3.98/4.00 [102]
	Los-loop	5min	15/30/45/60min	–	5.12/6.05/6.70/7.26 [102]
	LOOP	5min	5min	6.01% [105]	4.63 [105]
Demand	Q-Traffic	15min	15/30/45/60/ 75/90/105/120min	4.52%/7.93%/8.89%/9.24%/ 9.43%/9.56%/9.69%/9.78% [73]	–
	NYC Taxi	30min	30min	–	8.38 [72]
	NYC Bike	60min	60min	21.00% [77]	4.51 [77]
	TaxiBJ	30min	30min	13.80% [77]	17.24 [77]
Travel time	Chengdu	–	–	11.89% [116]	–
Occupancy	PeMSD-SF	60min	7 rolling time windows (24 time-points at a time)	16.80% [84]	–

public data information used for the prediction task, which mainly consists of two parts: one is the public spatio-temporal sequence data commonly used in the prediction, and the other is the external data to improve the prediction accuracy. However, the latter data is not used by all models due to the design of different model frameworks or the availability of the data.

A. Public Datasets

Here, we list public, commonly used and large-scale real-world datasets in traffic prediction.

- **PeMS**: It is an abbreviation from the California Transportation Agency Performance Measurement System (PeMS), which is displayed on the map and collected in real-time by more than 39000 independent detectors. These sensors span the freeway system across all major metropolitan areas of the State of California. The source is available at: <http://pems.dot.ca.gov/>. Based on this system, several sub-dataset versions (PeMSD3/4/7(M)/7/8/-SF/-BAY) have appeared and are widely used. The main difference is the range of time and space, as well as the number of sensors included in the data collection.
- **PeMSD3**: This dataset is a piece of data processed by Song *et al.* It includes 358 sensors and flow information from 9/1/2018 to 11/30/2018. A processed version is available at: <https://github.com/Davidham3/STSGCN>.
- **PeMSD4**: It describes the San Francisco Bay Area, and contains 3848 sensors on 29 roads dated from 1/1/2018 until 2/28/2018, 59 days in total. A processed version is available at: <https://github.com/Davidham3/ASTGCN/tree/master/data/PEMS04>.
- **PeMSD7(M)**: It describes the District 7 of California containing 228 stations, and The time range

of it is in the weekdays of May and June of 2012. A processed version is available at: <https://github.com/Davidham3/STGCN/tree/master/datasets>.

PeMSD7: This version was publicly released by Song *et al.* It contains traffic flow information from 883 sensor stations, covering the period from 7/1/2016 to 8/31/2016. A processed version is available at: <https://github.com/Davidham3/STSGCN>.

PeMSD8: It depicts the San Bernardino Area, and contains 1979 sensors on 8 roads dated from 7/1/2016 until 8/31/2016, 62 days in total. A processed version is available at: <https://github.com/Davidham3/ASTGCN/tree/master/data/PEMS08>.

PeMSD-SF: This dataset describes the occupancy rate, between 0 and 1, of different car lanes of San Francisco bay area freeways. The time span of these measurements is from 1/1/2008 to 3/30/2009 and the data is sampled every 10 minutes. The source is available at: <http://archive.ics.uci.edu/ml/datasets/PEMS-SF>.

PeMSD-BAY: It contains 6 months of statistics on traffic speed, ranging from 1/1/2017 to 6/30/2017, including 325 sensors in the Bay area. The source is available at: <https://github.com/liyaguang/DCRNN>.

- **METR-LA**: It records four months of statistics on traffic speed, ranging from 3/1/2012 to 6/30/2012, including 207 sensors on the highways of Los Angeles County. The source is available at: <https://github.com/liyaguang/DCRNN>.
- **LOOP**: It is collected from loop detectors deployed on four connected freeways (I-5, I-405, I-90 and SR-520) in the Greater Seattle Area. It contains traffic state data from 323 sensor stations over the entirely

of 2015 at 5-minute intervals. The source is available at: <https://github.com/zhiyongc/Seattle-Loop-Data>.

- Los-loop: This dataset is collected in the highway of Los Angeles County in real time by loop detectors. It includes 207 sensors and its traffic speed is collected from 3/1/2012 to 3/7/2012. These traffic speed data is aggregated every 5 minutes. The source is available at: <https://github.com/lehaifeng/T-GCN/tree/master/data>.
- TaxiBJ: Trajectory data is the taxicab GPS data and meteorology data in Beijing from four time intervals: 1st Jul. 2013 - 30th Oct. 2013, 1st Mar. 2014 - 30th Jun. 2014, 1st Mar. 2015 - 30th Jun. 2015, 1st Nov. 2015 - 10th Apr. 2016. The source is available at: <https://github.com/lucktroy/DeepST/tree/master/data/TaxiBJ>.
- SZ-taxi: This is the taxi trajectory of Shenzhen from Jan.1 to Jan.31, 2015. It contains 156 major roads of Luohu District as the study area. The speed of traffic on each road is calculated every 15 minutes. The source is available at: <https://github.com/lehaifeng/T-GCN/tree/master/data>.
- NYC Bike: The bike trajectories are collected from NYC CitiBike system. There are about 13000 bikes and 800 stations in total. The source is available at: <https://www.citibikenyc.com/system-data>. A processed version is available at: <https://github.com/lucktroy/DeepST/tree/master/data/BikeNYC>.
- NYC Taxi: The trajectory data is taxi GPS data for New York City from 2009 to 2018. The source is available at: <https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page>.
- Q-Traffic dataset: It consists of three sub-datasets: query sub-dataset, traffic speed sub-dataset and road network sub-dataset. These data are collected in Beijing, China between April 1, 2017 and May 31, 2017, from the Baidu Map. The source is available at: <https://github.com/JingqingZ/BaiduTraffic#Dataset>.
- Chicago: This is the trajectory of shared bikes in Chicago from 2013 to 2018. The source is available at: <https://www.divvybikes.com/system-data>.
- BikeDC: It is taken from the Washington D.C. Bike System. The dataset includes data from 472 stations and four time intervals of 2011, 2012, 2014 and 2016. The source is available at: <https://www.capitalbikeshare.com/system-data>.
- ENG-HW: It contains traffic flow information from inter-city highways between three cities, recorded by British Government, with a time range of 2006 to 2014. The source is available at: <http://tris.highwaysengland.co.uk/detail/trafficflowdata>.
- T-Drive: It consists of tremendous amounts of trajectories of Beijing taxicabs from Feb.1st, 2015 to Jun. 2nd 2015. These trajectories can be used to calculate the traffic flow in each region. The source is available at: <https://www.microsoft.com/en-us/research/publication/t-drive-driving-directions-based-on-taxi-trajectories/>.
- I-80: It is collected detailed vehicle trajectory data on eastbound I-80 in the San Francisco Bay area in

Emeryville, CA, on April 13, 2005. The dataset is 45 minutes long, and the vehicle trajectory data provides the precise location of each vehicle in the study area every tenth of a second. The source is available at: <http://ops.fhwa.dot.gov/trafficanalysisstools/ngsim.htm>.

- DiDi chuxing: DiDi gaia data open program provides real and free desensitization data resources to the academic community. It mainly includes travel time index, travel and trajectory datasets of multiple cities. The source is available at: <https://gaia.didichuxing.com>.
- Travel Time Index data:

The dataset includes the travel time index of Shenzhen, Suzhou, Jinan, and Haikou, including travel time index and average driving speed of city-level, district-level, and road-level, and time range is from 1/1/2018 to 12/31/2018. It also includes the trajectory data of the Didi taxi platform from 10/1/2018 to 12/1/2018 in the second ring road area of Chengdu and Xi'an, as well as travel time index and average driving speed of road-level in the region, and Chengdu and Xi'an city-level. Moreover, the city-level, district-level, road-level travel time index and average driving speed of Chengdu and Xi'an from 1/1/2018 to 12/31/2018 is contained.

Travel data:

This dataset contains daily order data from 5/1/2017 to 10/31/2017 in Haikou City, including the latitude and longitude of the start and end of the order, as well as the order attribute of the order type, travel category, and number of passengers.

Trajectory data:

This dataset comes from the order driver trajectory data of the Didi taxi platform in October and November 2016 in the Second Ring Area of Xi'an and Chengdu. The trajectory point collection interval is 2-4s. The trajectory points have been processed for road binding, ensuring that the data corresponds to the actual road information. The driver and order information were encrypted, desensitized and anonymized.

B. Common External Data

Traffic prediction is often influenced by a number of complex factors, which are usually called external data. Here, we list common external data items.

- Weather condition: temperature, humidity, wind speed, visibility and weather state (sunny/rainy/windy/cloudy etc.)
- Driver ID:
Due to the different personal conditions of drivers, the prediction will have a certain impact, therefore, it is necessary to label the driver, and this information is mainly used for personal prediction.
- Event: It includes various holidays, traffic control, traffic accidents, sports events, concerts and other activities.
- Time information: day-of-week, time-of-day.
(1) day-of-week usually includes weekdays and weekends due to the distinguished properties.

TABLE V
OPEN SOURCE CODES OF COMPARISON METHODS

Approach	Link
STGCN [51]	https://github.com/VeritasYin/STGCN_IJCAI-18
DCRNN [53]	https://github.com/liyaguang/DCRNN
ASTGCN [67]	https://github.com/guoshnBJTU/ASTGCN-r-pytorch
Graph WaveNet [55]	https://github.com/nanzhan/Graph-WaveNet
STSGCN [56]	https://github.com/Davidham3/STSGCN
GMAN [62]	https://github.com/zhengchuanpan/GMAN

(2) time-of-day generally has two division methods, one is to empirically examine the distribution with respect to time in the training dataset, 24 hours in each day can be intuitively divided into 3 periods: peak hours, off-peak hours, and sleep hours. The other is to manually divide one day into several timeslots, each timeslot corresponds to an interval.

VI. EXPERIMENTAL ANALYSIS AND DISCUSSIONS

In this section, we conduct experimental studies for several deep learning based traffic prediction methods, to identify the key components in each model. To this end, we utilize METR-LA dataset for speed prediction, evaluate the state-of-the-art approaches with public codes on this dataset, and investigate the performance limits.

A. Experimental Setup

In the experiment, we compare the performance of six typical speed prediction methods with public codes on a public dataset. Table V summarizes the links of public source codes for related comparison methods.

METR-LA dataset: This dataset contains 207 sensors and collects 4 months of data ranging from Mar 1st 2012 to Jun 30th 2012 for the experiment. 70% of data is used for training, 20% is used for testing while the remaining 10% for validation. Traffic speed readings are aggregated into 5 minutes windows, and Z-Score is applied for normalization. To construct the road network graph, each traffic sensor is considered as a node, and the adjacency matrix of the nodes is constructed by road network distance with a thresholded Gaussian kernel [122].

We use the following three metrics to evaluate different models: Mean Absolute Error (MAE), Rooted Mean Squared Error (RMSE), and Mean Absolute Percent Error (MAPE).

$$MAE = \frac{1}{\xi} \sum_{i=1}^{\xi} |\hat{y}^i - y^i|, \quad (15)$$

$$RMSE = \sqrt{\frac{1}{\xi} \sum_{i=1}^{\xi} (\hat{y}^i - y^i)^2}, \quad (16)$$

$$MAPE = \frac{1}{\xi} \sum_{i=1}^{\xi} \left| \frac{\hat{y}^i - y^i}{y^i} \right| * 100\%, \quad (17)$$

where \hat{y}^i and y^i denote the predicted value and the ground truth of region i for predicted time step, and ξ is the total number of samples.

For hyperparameter settings in the comparison algorithms, we set their values according to the experiments in the corresponding literatures ([51], [53], [55], [56], [62], [67]).

B. Experimental Results and Analysis

In this section, we evaluate the performance of various advanced traffic speed prediction methods on the graph-structured data, and the prediction results in the next 15 minute, 30 minute, and 60 minute ($T = 3, 6, 12$) are shown in Table VI.

STGCN applied ChebNet graph convolution and 1D convolution to extract spatial dependencies and temporal correlations. ASTGCN leveraged two attention layers on the basis of STGCN to capture the dynamic correlations of traffic network in spatial dimension and temporal dimension, respectively. DCRNN was a cutting edge deep learning model for prediction, which used diffusion graph convolutional networks and RNN during training stage to learn the representations of spatial dependencies and temporal relations. Graph WaveNet combined graph convolution with dilated casual convolution to capture spatial-temporal dependencies. STSGCN simultaneously extracted localized spatio-temporal correlation information based on the adjacency matrix of localized spatio-temporal graph. GMAN used purely attention structures in spatial and temporal dimensions to model dynamic spatio-temporal correlations.

As can be seen from the experimental results in Table VI: First, the attention-based methods (GMAN) perform better than other GCN-based methods in extracting spatial correlations. When modeling spatial correlations, GCN uses sum, mean or max functions to aggregate the features of each node's neighbors, ignoring the relative importance of different neighbors. On the contrary, the attention mechanism introduces the idea of weighting to realize adaptive updating of nodes at different times according to the importance of neighbor information, leading to better results. Second, the performance of the spectral models (STGCN and ASTGCN) is generally lower than that of the spatial models (DCRNN, Graph WaveNet and STSGCN). In addition, the results of most methods are not significantly different for 15min, but with the increase of the predicted time length, the performance of the attention-based method (GMAN) is significantly better than other GCN-based methods. Since most existing methods predict traffic conditions in an iterative manner, and their performance may not be greatly affected in short-term predictions because all historical observations used for prediction are error-free. However, as long-term prediction has to produce the results conditioned on previous predictions, resulting in error accumulations and reducing the accuracy of prediction greatly. Since the attention mechanism can directly perform multi-step predictions, ground-truth historical observations can be used regardless of short-term or long-term predictions, without the need to use error-prone values. Therefore, the above observations suggest possible ways to improve the prediction accuracy. First, the attention mechanism can extract the spatial information of road network more effectively. Second, the spatial-based approaches are generally more efficient than the

TABLE VI
PERFORMANCE OF TRAFFIC SPEED PREDICTION ON METR-LA

	T	Metric	STGCN	DCRNN	ASTGCN	Graph WaveNet	STSGCN	GMAN
METR-LA	15min	MAE	2.88	2.77	4.86	2.69	3.01	2.77
		RMSE	5.74	5.38	9.27	5.15	6.69	5.48
		MAPE	7.62%	7.30%	9.21%	6.90%	7.27%	7.25%
	30min	MAE	3.47	3.15	5.43	3.07	3.42	3.07
		RMSE	7.24	6.45	10.61	6.22	7.93	6.34
		MAPE	9.57%	8.80%	10.13%	8.37%	8.49%	8.35%
	60min	MAE	4.59	3.60	6.51	3.53	4.09	3.40
		RMSE	9.40	7.59	12.52	7.37	9.65	7.21
		MAPE	12.70%	10.50%	11.64%	10.01%	10.44%	9.72%

TABLE VII
COMPUTATION COST ON METR-LA

Method	Computation time		Number of parameters
	Training(s/epoch)	Inference(s)	
STGCN	49.24	28.13	320143
DCRNN	775.31	56.09	372352
ASTGCN	570.34	25.51	262379
Graph WaveNet	234.82	11.89	309400
STSGCN	560.29	25.56	1921886
GMAN	—	—	900801

spectral-based approaches when working with GCN. Third, the attention mechanism is more effective to improving the performance of long-term prediction when modeling temporal correlation. It is worth mentioning that adding an external data component is also beneficial for performance when external data is available.

C. Computational Complexity

To evaluate the computation complexity, we compare the computation time and the number of parameters among these models on the METR-LA dataset. All the experiments are conducted on the Tesla K80 with 12GB memory, the batchsize of each method is uniformly set to 64, T is set to 12, and we report the average training time of one epoch. For inference, we compute the time cost on the validation data. The results are shown in Table VII. STGCN adopts fully convolutional structures so that it is the fastest in training, and DCRNN uses the recurrent structures, which are very time consuming. Compared to methods (e.g., STGCN, DCRNN, ASTGCN, STSGCN) that require iterative calculations to generate 12 predicted results, Graph WaveNet can predict 12 steps ahead of time in one run, thus requiring less time for inference. STSGCN integrates three graphs at different moments into one graph as the adjacency matrix, which greatly increases the number of model parameters. Since GMAN is a pure attention mechanism model that consists of multiple attention mechanisms, it is necessary to calculate the relation between pairs of multiple variables, so the number of parameters is also high. Note that, when calculating the computation time of GMAN, it displays “out of memory” on our device, due to the relatively complex design of the model.

VII. FUTURE DIRECTIONS

Although traffic prediction has made great progress in recent years, there are still many open challenges that have not been fully investigated. These issues need to be addressed in future work. In the following discussion, we will state some future directions for further researches.

- Few shot problem: Most existing solutions are data intensive. However, abnormal conditions (extreme weather, temporary traffic control, etc) are usually non-recurrent, it is difficult to obtain data, which makes the training sample size smaller and learning more difficult than that under normal traffic conditions. In addition, due to the uneven development level of different cities, many cities have the problem of insufficient data. However, sufficient data is usually a prerequisite for deep learning methods. One possible solution to this problem is to use transfer learning techniques to perform deep spatio-temporal prediction tasks across cities. This technology aims to effectively transfer knowledge from a data-rich source city to a data-scarce target city. Although recent approaches have been proposed ([70], [91], [123], [124]), these researches have not been thoroughly studied, such as how to design a high-quality mathematical model to match two regions, or how to integrate other available auxiliary data sources, etc., are still worth considering and investigating.
- Knowledge graph fusion: Knowledge graph is an important tool for knowledge integration. It is a complex relational network composed of a large number of concepts, entities, entity relations and attributes. Transportation domain knowledge is hidden in multi-source and massive traffic big data. The construction, learning and deep knowledge search of large-scale transportation knowledge graph can help to dig deeper traffic semantic information and improve the prediction performance.
- Long-term prediction: Existing traffic prediction methods are mainly based on short-to-medium-term prediction, and there are very few studies on long-term forecasting ([100], [125], [126]). Long-term prediction is more difficult due to the more complex spatio-temporal dependencies and more uncertain factors. For long-term prediction, historical information may not have as much impact on short-to-medium-term prediction methods, and it may need to consider additional supplementary information.

- **Multi-source data:** Sensors, such as loop detectors or cameras, are currently the mainstream devices for collecting traffic data. However, due to the expensive installation and maintenance costs of sensors, the data is sparse. At the same time, most existing technologies based on previous and current traffic conditions are not suited to real-world factors, such as traffic accidents. In the big data era, a large amount of data has been produced in the field of transportation. When predicting traffic conditions, we can consider using several different datasets. In fact, these data are highly correlated. For example, to improve the performance of traffic flow prediction, we can consider information such as road network structure, traffic volume data, points of interests (POIs), and populations in a city. Effective fusion of multiple data can fill in the missing data and improve the accuracy of prediction.
- **Real-time prediction:** The purpose of real-time traffic prediction is to conduct data processing and traffic condition assessment in a short time. However, due to the increase of data, model size and parameters, the running time of the algorithm is too long to guarantee the requirement of real-time prediction. The scarce real-time prediction currently found in the literature [127], it is a great challenge to design an effective lightweight neural network to reduce the amount of network computation and improve network speed up.
- **Interpretability of models:** Due to the complex structure, large amount of parameters, low algorithm transparency, for neural networks, it is well known to verify its reliability. Lack of interpretability may bring potential problems to traffic prediction. Considering the complex data types and representations of traffic data, designing an interpretable deep learning model is more challenging than other types of data, such as images and text. Although some previous work combined the state space model to increase the interpretability of the model ([82]–[85]), how to establish a more interpretable deep learning model of traffic prediction has not been well studied and is still a problem to be solved.
- **Benchmarking traffic prediction:** As the field grows, more and more models have been proposed, and these models are often presented in a similar way. It has been increasingly difficult to gauge the effectiveness of new traffic prediction methods and compare models in the absence of a standardized benchmark with consistent experimental settings and large datasets. In addition, the design of models is becoming more and more complex. Although ablation studies have been done in most methods, it is still not clear how each component improves the algorithm. Therefore, it is of great importance to design a reproducible benchmarking framework with a standard common dataset.
- **High dimensionality.** At present, traffic prediction still mainly stays at the level of a single data source, with less consideration of influencing factors. With more collected datasets, we can obtain more influencing factors. However, high-dimensional features often bring about “curse of dimensionality” and high computational costs.

Therefore, how to extract the key factors from the large amount of influencing factors is an important issue to be resolved.

- **Prediction under perturbation.** In the process of collecting traffic data, due to factors such as equipment failures, the collected information deviates from the true value. Therefore, the actual sampled data is generally subject to noise pollution to varying degrees. The use of contaminated data for modeling will affect the prediction accuracy of the model. Existing methods usually treat data processing and model prediction as two separate tasks. It is of great practical significance to design a robust and effective traffic prediction model in the case of various noises and errors in the data.
- **The optimal network architecture choice:** For a given traffic prediction task, how to choose a suitable network architecture has not been well studied. For example, some works model the historical traffic data of each road as a time series and use networks such as RNN for prediction; some works model the traffic data of multiple roads as 2D spatial maps and use networks such as CNN to make predictions. In addition, some works model traffic data as a road network graph, so network architectures such as GNN are adopted. There is still a lack of more in-depth research on how to optimally choose a deep learning network architecture to better solve the prediction task studied.

VIII. CONCLUSION

In this paper, we conduct a comprehensive survey of various deep learning architectures for traffic prediction. More specifically, we first summarize the existing traffic prediction methods, and give a taxonomy of them. Then, we list the representative results in different traffic prediction tasks, comprehensively provide public available traffic datasets, and conduct a series of experiments to investigate the performance of existing traffic prediction methods. Finally, some major challenges and future research directions are discussed. This paper is suitable for participants to quickly understand the traffic prediction, so as to find branches they are interested in. It also provides a good reference and inquiry for researchers in this field, which can facilitate the relevant research.

REFERENCES

- [1] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi, “DNN-based prediction model for spatio-temporal data,” in *Proc. 24th ACM SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, Oct. 2016, pp. 1–4.
- [2] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, “Short-term traffic forecasting: Where we are and where we’re going,” *Transp. Res. C, Emerg. Technol.*, vol. 43, pp. 3–19, Jun. 2014.
- [3] Y. Li and C. Shahabi, “A brief overview of machine learning methods for short-term traffic forecasting and future directions,” *SIGSPATIAL Special*, vol. 10, no. 1, pp. 3–9, Jun. 2018, doi: 10.1145/3231541.3231544.
- [4] A. M. Nagy and V. Simon, “Survey on traffic prediction in smart cities,” *Pervas. Mobile Comput.*, vol. 50, pp. 148–163, Oct. 2018.
- [5] A. Singh, A. Shadan, R. Singh, and Ranjeet, “Traffic forecasting,” *Int. J. Sci. Res. Rev.*, vol. 7, no. 3, pp. 1565–1568, 2019.
- [6] A. Boukerche and J. Wang, “Machine learning-based traffic prediction models for intelligent transportation systems,” *Comput. Netw.*, vol. 181, Nov. 2020, Art. no. 107530.

- [7] I. Lana, J. D. Ser, M. Velez, and E. I. Vlahogianni, "Road traffic forecasting: Recent advances and new challenges," *IEEE Intell. Transp. Syst. Mag.*, vol. 10, no. 2, pp. 93–109, Apr. 2018.
- [8] D. A. Tedjopurnomo, Z. Bao, B. Zheng, F. Choudhury, and A. K. Qin, "A survey on modern deep neural network for traffic prediction: Trends, methods and challenges," *IEEE Trans. Knowl. Data Eng.*, early access, Jun. 9, 2020, doi: [10.1109/TKDE.2020.3001195](https://doi.org/10.1109/TKDE.2020.3001195).
- [9] J. Ye, J. Zhao, K. Ye, and C. Xu, "How to build a graph-based deep learning architecture in traffic domain: A survey," 2020, *arXiv:2005.11691*. [Online]. Available: <http://arxiv.org/abs/2005.11691>
- [10] P. Xie, T. Li, J. Liu, S. Du, X. Yang, and J. Zhang, "Urban flow prediction from spatiotemporal data using machine learning: A survey," *Inf. Fusion*, vol. 59, pp. 1–12, Jul. 2020, doi: [10.1016/j.inffus.2020.01.002](https://doi.org/10.1016/j.inffus.2020.01.002).
- [11] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results," *J. Transp. Eng.*, vol. 129, no. 6, pp. 664–672, Nov. 2003.
- [12] E. Zivot and J. Wang, "Vector autoregressive models for multivariate time series," in *Modeling Financial Time Series With S-PLUS*. New York, NY, USA: Springer, 2006, pp. 385–429.
- [13] R. Chen, C.-Y. Liang, W.-C. Hong, and D.-X. Gu, "Forecasting holiday daily tourist flow based on seasonal support vector regression with adaptive genetic algorithm," *Appl. Soft Comput.*, vol. 26, pp. 435–443, Jan. 2015.
- [14] U. Johansson, H. Boström, T. Löfström, and H. Linusson, "Regression conformal prediction with random forests," *Mach. Learn.*, vol. 97, nos. 1–2, pp. 155–176, 2014.
- [15] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.
- [16] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [17] N. Kalchbrenner and P. Blunsom, "Recurrent continuous translation models," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2013, pp. 1700–1709.
- [18] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 1–14.
- [19] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.
- [20] J. L. Elman, "Distributed representations, simple recurrent networks, and grammatical structure," *Mach. Learn.*, vol. 7, nos. 2–3, pp. 195–225, Sep. 1991.
- [21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [22] K. Cho *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, *arXiv:1406.1078*. [Online]. Available: <http://arxiv.org/abs/1406.1078>
- [23] S. Shekhar and B. M. Williams, "Adaptive seasonal time series models for forecasting short-term traffic flow," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 2024, no. 1, pp. 116–125, Jan. 2007.
- [24] X. Li, G. Pan, Z. Wu, G. Qi, S. Li, D. Zhang, W. Zhang, and Z. Wang, "Prediction of urban human mobility using large-scale taxi traces and its applications," *Frontiers Comput. Sci.*, vol. 6, no. 1, pp. 111–121, 2012.
- [25] L. Moreira-Matias, J. Gama, M. Ferreira, J. Mendes-Moreira, and L. Damas, "Predicting taxi-passenger demand using streaming data," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1393–1402, Sep. 2013.
- [26] M. Lippi, M. Bertini, and P. Frasconi, "Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 2, pp. 871–882, Jun. 2013.
- [27] I. Wagner-Muns, I. Guardiola, V. Samaranayake, and W. Kayani, "A functional data analysis approach to traffic volume forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 878–888, Jun. 2017.
- [28] W. Li *et al.*, "A general framework for unmet demand prediction in on-demand transport services," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 8, pp. 2820–2830, Aug. 2019.
- [29] J. Guan, W. Wang, W. Li, and S. Zhou, "A unified framework for predicting KPIs of on-demand transport services," *IEEE Access*, vol. 6, pp. 32005–32014, 2018.
- [30] L. Tang, Y. Zhao, J. Cabrera, J. Ma, and K. L. Tsui, "Forecasting short-term passenger flow: An empirical study on Shenzhen metro," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3613–3622, Oct. 2018.
- [31] Z. Diao *et al.*, "A hybrid model for short-term traffic volume prediction in massive transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 935–946, Mar. 2019.
- [32] D. Salinas, M. Bohlke-Schneider, L. Callot, R. Medico, and J. Gasthaus, "High-dimensional multivariate forecasting with low-rank Gaussian copula processes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 6824–6834.
- [33] L. Lin, J. Li, F. Chen, J. Ye, and J. Huai, "Road traffic speed prediction: A probabilistic model fusing multi-source data," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 7, pp. 1310–1323, Jul. 2018.
- [34] P. Duan, G. Mao, W. Liang, and D. Zhang, "A unified spatio-temporal model for short-term traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 9, pp. 3212–3223, Sep. 2019.
- [35] H. Tan, Y. Wu, B. Shen, P. J. Jin, and B. Ran, "Short-term traffic prediction based on dynamic tensor completion," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2123–2133, Aug. 2016.
- [36] J. Shin and M. Sunwoo, "Vehicle speed prediction using a Markov chain with speed constraints," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 9, pp. 3201–3211, Sep. 2019.
- [37] K. Ishibashi, S. Harada, and R. Kawahara, "Inferring latent traffic demand offered to an overloaded link with modeling QoS-degradation effect," *IEICE Trans. Commun.*, vol. E102.B, no. 4, pp. 790–798, Apr. 2019.
- [38] Y. Gong, Z. Li, J. Zhang, W. Liu, Y. Zheng, and C. Kirsch, "Network-wide crowd flow prediction of Sydney trains via customized online non-negative matrix factorization," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2018, pp. 1243–1252.
- [39] N. Polson and V. Sokolov, "Bayesian particle tracking of traffic flows," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 345–356, Feb. 2018.
- [40] H. Hong *et al.*, "Learning common metrics for homogenous tasks in traffic flow prediction," in *Proc. IEEE 14th Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2015, pp. 1007–1012.
- [41] H. Yu, N. Rao, and I. Dhillon, "Temporal regularized matrix factorization for high-dimensional time series prediction," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 847–855.
- [42] D. Deng, C. Shahabi, U. Demiryurek, L. Zhu, R. Yu, and Y. Liu, "Latent space model for road networks to predict time-varying traffic," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1525–1534.
- [43] D. Deng, C. Shahabi, U. Demiryurek, and L. Zhu, "Situation aware multi-task learning for traffic prediction," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2017, pp. 81–90.
- [44] A. Kinoshita, A. Takasu, and J. Adachi, "Latent variable model for weather-aware traffic state analysis," in *Proc. Int. Workshop Inf. Search, Integr., Personalization*. Lyon, France: Springer, 2016, pp. 51–65.
- [45] V. I. Shvetsov, "Mathematical modeling of traffic flows," *Autom. Remote Control*, vol. 64, no. 11, pp. 1651–1689, 2003.
- [46] J. M. Chiou, "Dynamic functional prediction and classification, with application to traffic flow prediction," *Oper. Res.*, vol. 53, no. 3, pp. 239–241, 2013.
- [47] Y. Gong, Z. Li, J. Zhang, W. Liu, and J. Yi, "Potential passenger flow prediction: A novel study for urban transportation development," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 4020–4027.
- [48] Z. Li, N. Sergin, H. Yan, C. Zhang, and F. Tsung, "Tensor completion for weakly-dependent data on graph for metro passenger flow prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 4804–4810.
- [49] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3844–3852.
- [50] T. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–14.
- [51] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 3634–3640.
- [52] X. Geng *et al.*, "Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 3656–3663.
- [53] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural networks: Data-driven traffic flow forecasting," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–16.
- [54] C. Chen *et al.*, "Gated residual recurrent graph neural networks for traffic prediction," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 485–492.

- [55] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph WaveNet for deep spatial-temporal graph modeling," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 1907–1913.
- [56] C. Song, Y. Lin, S. Guo, and H. Wan. (2020). *Spatial-Temporal Synchronous Graph Convolutional Networks: A New Framework for Spatial-Temporal Network Data Forecasting*. [Online]. Available: <https://github.com/wanhuiyu/STSGCN/blob/master/paper/AAAI2020-STSGCN.pdf>
- [57] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*. [Online]. Available: <http://arxiv.org/abs/1409.0473>
- [58] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, and J. Zhang, "Urban traffic prediction from spatio-temporal data using deep meta learning," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 1720–1730.
- [59] Y. Li, Z. Zhu, D. Kong, M. Xu, and Y. Zhao, "Learning heterogeneous spatial-temporal representation for bike-sharing demand prediction," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 1004–1011.
- [60] J. Zhang, X. Shi, J. Xie, H. Ma, I. King, and D.-Y. Yeung, "GaAN: Gated attention networks for learning on large and spatiotemporal graphs," 2018, *arXiv:1803.07294*. [Online]. Available: <http://arxiv.org/abs/1803.07294>
- [61] Y. Li and J. M. F. Moura, "Forecaster: A graph transformer for forecasting spatial and time-dependent data," 2019, *arXiv:1909.04019*. [Online]. Available: <http://arxiv.org/abs/1909.04019>
- [62] C. Zheng, X. Fan, C. Wang, and J. Qi, "GMAN: A graph multi-attention network for traffic prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 1234–1241.
- [63] C. Park *et al.*, "ST-GRAT: A novel spatio-temporal graph attention network for accurately forecasting dynamically changing road speed," 2019, *arXiv:1911.13181*. [Online]. Available: <http://arxiv.org/abs/1911.13181>
- [64] X. Geng *et al.* (2020). *CGT: Clustered Graph Transformer for Urban Spatio-Temporal Prediction*. [Online]. Available: <https://openreview.net/forum?id=H1eJAANtvr>
- [65] X. Shi, H. Qi, Y. Shen, G. Wu, and B. Yin, "A spatial-temporal attention approach for traffic prediction," *IEEE Trans. Intell. Transp. Syst.*, early access, Apr. 13, 2020, doi: [10.1109/TITS.2020.2983651](https://doi.org/10.1109/TITS.2020.2983651).
- [66] X. Yin, G. Wu, J. Wei, Y. Shen, H. Qi, and B. Yin, "Multi-stage attention spatial-temporal graph networks for traffic prediction," *Neurocomputing*, vol. 428, pp. 42–53, Mar. 2021.
- [67] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 922–929.
- [68] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. Dauphin, "Convolutional sequence to sequence learning," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 1243–1252.
- [69] S. Fang, Q. Zhang, G. Meng, S. Xiang, and C. Pan, "GSTNet: Global spatial-temporal network for traffic flow prediction," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 10–16.
- [70] H. Yao, Y. Liu, Y. Wei, X. Tang, and Z. Li, "Learning from multiple cities: A meta-learning approach for spatial-temporal prediction," in *Proc. World Wide Web Conf. (WWW)*, 2019, pp. 2181–2191.
- [71] L. Zhu and N. Laptev, "Deep and confident prediction for time series at Uber," in *Proc. IEEE Int. Conf. Data Mining Workshops (ICDMW)*, Nov. 2017, pp. 103–110.
- [72] J. Ye, L. Sun, B. Du, Y. Fu, X. Tong, and H. Xiong, "Co-prediction of multiple transportation demands based on deep spatio-temporal neural network," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 305–313.
- [73] B. Liao *et al.*, "Deep sequence learning with auxiliary information for traffic prediction," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 537–546.
- [74] R. Jiang *et al.*, "Deepurbanevent: A system for predicting citywide crowd dynamics at big events," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2019, pp. 2114–2122.
- [75] Z. Zhang, M. Li, X. Lin, Y. Wang, and F. He, "Multistep speed prediction on traffic networks: A graph convolutional Sequence-to-Sequence learning approach with attention mechanism," 2018, *arXiv:1810.10237*. [Online]. Available: <http://arxiv.org/abs/1810.10237>
- [76] Y. Wang, H. Yin, H. Chen, T. Wo, J. Xu, and K. Zheng, "Origin-destination matrix prediction via graph convolution: A new perspective of passenger demand modeling," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2019, pp. 1227–1235.
- [77] L. Bai, L. Yao, S. S. Kanhere, X. Wang, and Q. Z. Sheng, "STG2Seq: Spatial-temporal graph to sequence model for multi-step passenger demand forecasting," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 1981–1987.
- [78] P. Deshpande and S. Sarawagi, "Streaming adaptation of deep forecasting models using adaptive recurrent units," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 1560–1568.
- [79] D. Chai, L. Wang, and Q. Yang, "Bike flow prediction with multi-graph convolutional networks," in *Proc. 26th ACM SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, Nov. 2018, pp. 397–400.
- [80] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li, "Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 5668–5675.
- [81] X. Fang, J. Huang, F. Wang, L. Zeng, H. Liang, and H. Wang, "ConSTGAT: Contextual spatial-temporal graph attention network for travel time estimation at Baidu maps," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 2697–2705.
- [82] L. Li, J. Yan, X. Yang, and Y. Jin, "Learning interpretable deep state space model for probabilistic time series forecasting," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 2901–2908.
- [83] Z. Pan, Z. Wang, W. Wang, Y. Yu, J. Zhang, and Y. Zheng, "Matrix factorization for spatio-temporal neural networks with applications to urban flow prediction," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, Nov. 2019, pp. 2683–2691.
- [84] R. Sen, H. Yu, and I. Dhillon, "Think globally, act locally: A deep neural network approach to high-dimensional time series forecasting," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 4838–4847.
- [85] A. Ziat, E. Delasalles, L. Denoyer, and P. Gallinari, "Spatio-temporal neural networks for space-time series forecasting and relations discovery," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2017, pp. 705–714.
- [86] Z. Lin, J. Feng, Z. Lu, Y. Li, and D. Jin, "DeepSTN+: Context-aware spatial-temporal neural network for crowd flow prediction in metropolis," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 1020–1027.
- [87] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 1655–1661.
- [88] J. Zhang, Y. Zheng, J. Sun, and D. Qi, "Flow prediction in spatio-temporal networks based on multitask deep learning," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 3, pp. 468–478, Mar. 2020.
- [89] S. Guo, Y. Lin, S. Li, Z. Chen, and H. Wan, "Deep spatial-temporal 3D convolutional neural networks for traffic data forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3913–3926, Oct. 2019.
- [90] T. Li, J. Zhang, K. Bao, Y. Liang, Y. Li, and Y. Zheng, "AutoST: Efficient neural architecture search for spatio-temporal prediction," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 794–802.
- [91] L. Wang, X. Geng, X. Ma, F. Liu, and Q. Yang, "Cross-city transfer learning for deep spatio-temporal prediction," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 1893–1899.
- [92] J. Zhao, T. Zhu, R. Zhao, and P. Zhao, "Layerwise recurrent autoencoder for general real-world traffic flow forecasting," in *Proc. 9th Int. Conf. Intell. Sci. Big Data Eng., Big Data Mach. Learn.*, 2019, pp. 78–88.
- [93] A. Zonoozi, J.-J. Kim, X.-L. Li, and G. Cong, "Periodic-CRN: A convolutional recurrent model for crowd density prediction with recurring periodic patterns," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 3732–3738.
- [94] Z. Zheng, Y. Yang, J. Liu, H.-N. Dai, and Y. Zhang, "Deep and embedded learning approach for traffic flow prediction in urban informatics," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3927–3939, Oct. 2019.
- [95] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, "Adaptive graph convolutional recurrent network for traffic forecasting," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020.
- [96] K. Guo *et al.*, "Optimized graph convolution recurrent neural network for traffic prediction," *IEEE Trans. Intell. Transp. Syst.*, early access, Jan. 14, 2020, doi: [10.1109/TITS.2019.2963722](https://doi.org/10.1109/TITS.2019.2963722).
- [97] Z. Liu, F. Miranda, W. Xiong, J. Yang, Q. Wang, and C. T. Silva, "Learning geo-contextual embeddings for commuting flow prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 808–816.
- [98] X. Wang, C. Chen, Y. Min, J. He, B. Yang, and Y. Zhang, "Efficient metropolitan traffic prediction based on graph recurrent neural network," 2018, *arXiv:1811.00740*. [Online]. Available: <http://arxiv.org/abs/1811.00740>

- [99] X. Tang, H. Yao, Y. Sun, C. Aggarwal, P. Mitra, and S. Wang, "Joint modeling of local and global temporal dynamics for multivariate time series forecasting with missing values," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 5956–5963.
- [100] D. Zang, J. Ling, Z. Wei, K. Tang, and J. Cheng, "Long-term traffic speed prediction based on multiscale spatio-temporal feature learning network," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3700–3709, Oct. 2019.
- [101] Z. Lv, J. Xu, K. Zheng, H. Yin, P. Zhao, and X. Zhou, "LC-RNN: A deep learning model for traffic speed prediction," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 3470–3476.
- [102] L. Zhao *et al.*, "T-GCN: A temporal graph convolutional network for traffic prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3848–3858, Sep. 2020.
- [103] Z. Diao, X. Wang, D. Zhang, Y. Liu, K. Xie, and S. He, "Dynamic spatial-temporal graph convolutional neural networks for traffic forecasting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 890–897.
- [104] N. Zhang, X. Guan, J. Cao, X. Wang, and H. Wu, "A hybrid traffic speed forecasting approach integrating wavelet transform and motif-based graph convolutional recurrent neural network," 2019, *arXiv:1904.06656*. [Online]. Available: <http://arxiv.org/abs/1904.06656>
- [105] Z. Cui, K. Henrickson, R. Ke, and Y. Wang, "Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, pp. 4883–4894, Nov. 2020.
- [106] R. Huang, C. Huang, Y. Liu, G. Dai, and W. Kong, "LSGCN: Long short-term traffic prediction with graph convolutional networks," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, Jul. 2020, pp. 2355–2361.
- [107] W. Chen, L. Chen, Y. Xie, W. Cao, Y. Gao, and X. Feng, "Multi-range attentive bicomponent graph convolutional network for traffic forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 3529–3536.
- [108] J. Xu, R. Rahmatizadeh, L. Bölöni, and D. Turgut, "Real-time prediction of taxi demand using recurrent neural networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 8, pp. 2572–2581, Aug. 2018.
- [109] D. Lee, S. Jung, Y. Cheon, D. Kim, and S. You, "Forecasting taxi demands with fully convolutional networks and temporal guided embedding," in *Proc. 32nd Conf. Neural Inf. Process. Syst. Workshop Modeling Decis.-Making Spatiotemporal Domain*, 2018.
- [110] L. Liu, Z. Qiu, G. Li, Q. Wang, W. Ouyang, and L. Lin, "Contextualized spatial-temporal network for taxi origin-destination demand prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3875–3887, Oct. 2019.
- [111] H. Yao *et al.*, "Deep multi-view spatial-temporal network for taxi demand prediction," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 2588–2595.
- [112] J. Ke, H. Zheng, H. Yang, and X. M. Chen, "Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach," *Transp. Res. C, Emerg. Technol.*, vol. 85, pp. 591–608, Dec. 2017.
- [113] N. Davis, G. Raina, and K. Jagannathan, "Grids versus graphs: Partitioning space for improved taxi demand-supply forecasts," 2019, *arXiv:1902.06515*. [Online]. Available: <http://arxiv.org/abs/1902.06515>
- [114] J. Pang, J. Huang, Y. Du, H. Yu, Q. Huang, and B. Yin, "Learning to predict bus arrival time from heterogeneous measurements via recurrent neural network," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 9, pp. 3283–3293, Sep. 2019.
- [115] P. He, G. Jiang, S.-K. Lam, and D. Tang, "Travel-time prediction of bus journey with multiple bus trips," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 11, pp. 4192–4205, Nov. 2019.
- [116] D. Wang, J. Zhang, W. Cao, J. Li, and Y. Zheng, "When will you arrive? Estimating travel time based on deep neural networks," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 2500–2507.
- [117] R. Dai, S. Xu, Q. Gu, C. Ji, and K. Liu, "Hybrid spatio-temporal graph convolutional network: Improving traffic prediction with navigation data," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2020, pp. 3074–3082.
- [118] G. Lai, W. Chang, Y. Yang, and H. Liu, "Modeling long-and short-term temporal patterns with deep neural networks," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2018, pp. 95–104.
- [119] AMAP Technology. (2019). *AMAP Technology Annual in 2019*. [Online]. Available: <https://files.alicdn.com/tppservice/46a2ae997f5ef395a78c9ab751b6d942.pdf>
- [120] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 156–165.
- [121] X. Zhang, L. Xie, Z. Wang, and J. Zhou, "Boosted trajectory calibration for traffic state estimation," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2019, pp. 866–875.
- [122] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.
- [123] Y. Wei, Y. Zheng, and Q. Yang, "Transfer knowledge between cities," in *Proc. 22nd SIGKDD Conf. Knowl. Discovery Data Mining*, 2016, pp. 1905–1914.
- [124] E. L. Manibardo, I. Laña, and J. D. Ser, "Transfer learning and online learning for traffic forecasting under different data availability conditions: Alternatives and pitfalls," 2020, *arXiv:2005.05069*. [Online]. Available: <http://arxiv.org/abs/2005.05069>
- [125] I. Laña, J. L. Lobo, E. Capecci, J. D. Ser, and N. Kasabov, "Adaptive long-term traffic state estimation with evolving spiking neural networks," *Transp. Res. C, Emerg. Technol.*, vol. 101, pp. 126–144, Apr. 2019.
- [126] Z. Wang, X. Su, and Z. Ding, "Long-term traffic prediction based on LSTM encoder-decoder architecture," *IEEE Trans. Intell. Transp. Syst.*, early access, Jun. 3, 2020, doi: [10.1109/TITS.2020.2995546](https://doi.org/10.1109/TITS.2020.2995546).
- [127] E. L. Manibardo, I. Laña, J. L. Lobo, and J. D. Ser, "New perspectives on the use of online learning for congestion level prediction over traffic data," 2020, *arXiv:2003.14304*. [Online]. Available: <http://arxiv.org/abs/2003.14304>



Xuayan Yin received the B.S. degree from the Changchun University of Technology, in 2013, and the M.E. degree from Northeast Normal University, in 2018. She is currently pursuing the Ph.D. degree with the Dalian University of Technology. Her research interests include spatio-temporal data mining and sequential prediction.



Genze Wu is currently pursuing the bachelor's degree in computer science with the Dalian University of Technology. His research interests include data mining and computer vision.



Jinze Wei received the B.E. degree in mechanical engineering from the Dalian University of Technology, in 2018. He is currently pursuing the master's degree with the School of Computer Science and Technology, Dalian University of Technology. His research interests include data mining and intelligent transportation technology.



Yanming Shen received the B.S. degree in automation from Tsinghua University in 2000 and the Ph.D. degree from the Department of Electrical and Computer Engineering, Polytechnic University (now NYU Tandon School of Engineering) in 2007. He is currently a Professor with the School of Computer Science and Technology, Dalian University of Technology, China. His general research interests include big data analytics, distributed systems, and networking. He was a recipient of the 2011 Best Paper Award for Multimedia Communications (awarded by IEEE Communications Society).



Heng Qi (Member, IEEE) received the B.S. degree from Hunan University in 2004 and the M.E. and Ph.D. degrees from the Dalian University of Technology, in 2006 and 2012, respectively. He has been a JSPS overseas research fellow with the Graduate School of Information Science, Nagoya University, Japan, from 2016 to 2017. He is currently an Associate Professor with the School of Computer Science and Technology, Dalian University of Technology, China. His research interests include computer networks and multimedia computing.



Baocai Yin (Member, IEEE) received the M.S. and Ph.D. degrees in computational mathematics from the Dalian University of Technology, Dalian, China, in 1988 and 1993, respectively. He is currently a Professor of computer science and technology with the Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology. He is also a Researcher with the Beijing Key Laboratory of Multimedia and Intelligent Software Technology and the Beijing Advanced Innovation Center for Future Internet Technology. He has authored or coauthored more than 200 academic articles in prestigious international journals, including the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON CYBERNETICS, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and top-level conferences, such as CVPR, AAAI, INFOCOM, IJCAI, and ACM SIGGRAPH. His research interests include multimedia, image processing, computer vision, and pattern recognition.