

# A Comprehensive Survey on Traffic Prediction

Xueyan Yin, Genze Wu, Jinze Wei, Yanming Shen, Heng Qi, and Baocai Yin

**Abstract**—Traffic prediction plays an essential role in intelligent transportation system. Accurate traffic prediction can assist route planning, 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, greatly advancing traffic prediction abilities. The purpose of this paper is to provide a comprehensive survey for traffic prediction. Specifically, we first summarize the existing traffic prediction methods, and give a taxonomy of them. Second, we list the common tasks of traffic prediction and the state-of-the-art in these tasks. Third, we collect and organize widely used public datasets in the existing literature. Furthermore, we give an evaluation by conducting extensive experiments to compare the performance of different methods related to traffic demand and speed prediction respectively on two datasets. Finally, we discuss potential future directions.

**Index Terms**—Traffic Prediction, Machine Learning, Deep Learning.

## 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 the cornerstone of the development of ITS. Accurate traffic prediction is essential to many real-world applications. For example, traffic volume 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.

**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. The specific expression is as follows:

- **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 is sometimes more correlated with the predicted time step than

X. Yin, G. Wu, J. Wei, Y. Shen, H. Qi and B. Yin are with the School of Electronic Information and Electrical Engineering, Dalian University of Technology, China, 116024. Email: {yinxueyan, ggwdwsbs, wei-jinze}@mail.dlut.edu.cn, and {shen, hengqi, ybc}@dlut.edu.cn.

that of the near 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.

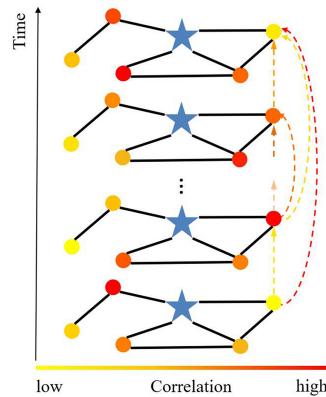


Fig. 1. Complex spatio-temporal correlations. The nodes represent different locations in the road network, and the blue star node represents the predicted target. The darker the color, the greater the spatial correlation with the target node. The dotted line shows the temporal correlation between different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

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

To sum up, traffic data shows strong dynamic correlation in both spatial and temporal dimensions. Therefore, how to mine the non-linear and complicated spatial-temporal patterns so as to make accurate traffic prediction is an important topic. 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 timestamp, 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. Usually, the demand for traffic prediction includes the demand for taxis and shared bikes.

- **Occupancy**

The occupancy rate explains the extent to which vehicles occupy road space. It also considers changes in traffic composition and speed when measuring, and provides a more reliable indicator of the extent to which vehicles occupy a road.

- **Travel time**

In the case of obtaining the route of any two points in the road network, estimating the travel time is required to get from one point to another in the route. In general, the travel time should include the waiting time at the intersection.

**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] outlined the significance and research directions of traffic prediction. [5] 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 still lacks of a broad and systematic survey on exploring traffic prediction in general.

**Our contributions** To our knowledge, this is the first comprehensive survey to review the state-of-the-art in traffic prediction from multiple perspectives including approaches, applications, datasets, and experiments. 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.

**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: **traditional methods** and **deep learning-based methods**. Traditional methods include classical statistical methods and machine learning methods. The classical statistical method is to build a statistical model based on data to predict and analyze the data. The most representative and common algorithms are Historical Average (HA), Auto-Regressive Integrated Moving Average (ARIMA)

[6], and Vector Auto-Regressive (VAR) [7]. 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, so their performance is usually poor in practical applications. Later, a number of machine learning methods, such as Support Vector Regression (SVR) [8] and Random Forest Regression (RFR) [9], were proposed for traffic prediction problem. Such methods have the ability to process high-dimensional data and capture complex non-linear relationships. However, the performance of these studies is still limited in mining complex spatio-temporal patterns because they require additional hand-crafted features designed in advance by domain experts, which often do not fully describe the properties of the data, rather than learning directly from the raw data.

It was not until the advent of deep learning-based methods that the full potential of artificial intelligence in traffic prediction was developed [10]. This technology studies how to learn a hierarchical model to map the original input directly to the expected output [11]. 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 Networks (CNN) [12] is employed to extract spatial correlation of the grid-structured data described by images or videos, and Graph Convolutional Network (GCN) [13] extends convolution operation to more general graph-structured data, which is more suitable to represent the traffic network structure. Furthermore, Recurrent Neural Network (RNN) [14], [15] and its variants LSTM [16] or GRU [17] are commonly utilized to model temporal dependency. Here, we summarize the two categories and categorize each in more details, as shown in Fig. 2.

**Organization of this survey** The rest of this paper is organized as follows. Section II covers the traditional 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. Section IV lists the state-of-the-art 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 in two common scenarios. Section VII discusses several significant and important directions of future traffic prediction problem and gives some open problems. Finally, we conclude this paper in Section VIII.

## II. TRADITIONAL METHODS

Classical statistical and machine learning models are two major representative data-driven methods for traffic prediction. In time-series analysis, autoregressive integrated moving average (ARIMA) [6] and its variants are one of the most consolidated approaches based on classical statistics and have been widely applied for traffic prediction problems ([6], [18]–[22]). However, these methods are generally designed for

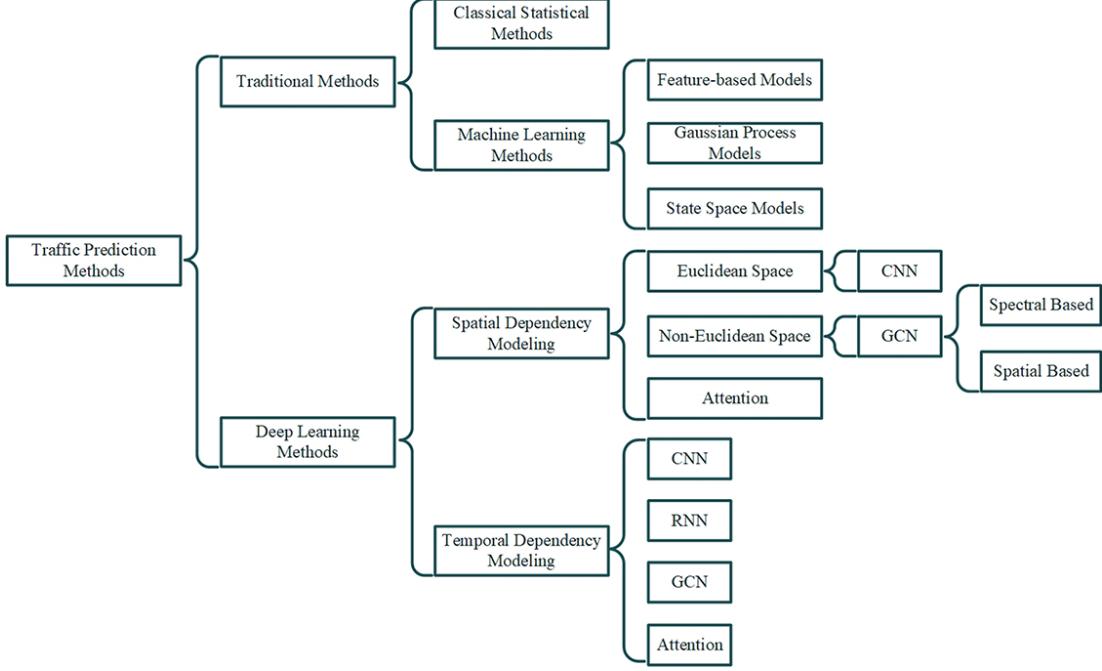


Fig. 2. Categories of traffic prediction methods.

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.

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 ([23], [24]) 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. Despite this feasibility, feature-based models have a crucial limitation: the performance of the model depends heavily on the human-engineered features. Gaussian process models model 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 ([25]–[27]), they 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 ([28]–[38]) is limited, and most of the time they are not optimal for modeling complex and dynamic traffic data. Table I summarizes some recent representative traditional approaches.

### III. DEEP LEARNING METHODS

Deep learning models exploit much more features and complex architectures than the traditional methods, and can

achieve better performance. They have been widely applied in traffic prediction. In this section, we will review different deep learning based traffic prediction methods in recent years according to how they model spatio-temporal correlations.

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

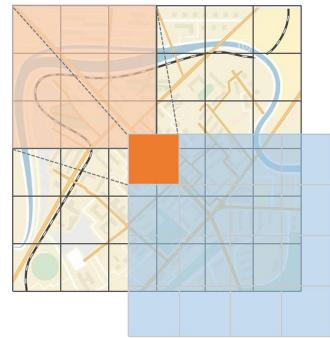


Fig. 3. 2D Convolution. Each grid in the image is treated as a region, where neighbors are determined by the filter size. The 2D convolution operates between a certain region and its neighbors. The neighbors of a region are ordered and have a fixed size.

As shown in Fig. 3, each region is directly connected to its nearby regions. With a  $3 \times 3$  window, the neighborhood of each

TABLE I  
TRADITIONAL METHODS.

Category		Approach
Statistical methods		[6], [18]–[22]
Machine learning methods	Feature-based models	[23], [24]
	Gaussian process models	[25]–[27]
	State space models	[28]–[40]

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  $3 \times 3$  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. [13] 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:

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

where  $\mathbf{U}$  is the matrix of eigenvectors of normalized graph Laplacian  $\mathbf{L}$ , which is defined as  $\mathbf{L} = \mathbf{I}_N - \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} = \mathbf{U} \Lambda \mathbf{U}^T$ ,  $\mathbf{D}$  is the diagonal matrix,  $\mathbf{D}_{ii} = \sum_j (\mathbf{A}_{ij})$ ,  $\mathbf{A}$  is the adjacency matrix of the graph,  $\Lambda$  is the diagonal matrix of eigenvalues,  $\Lambda = \lambda_i$ . 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. [42] introduced a filter as Chebyshev polynomials of the diagonal matrix of eigenvalues, i.e,  $\mathbf{g}_\theta = \sum_{i=1}^K \theta_i \mathbf{T}_k(\tilde{\Lambda})$ , where  $\theta \in \mathbb{R}^K$  is now a vector of Chebyshev coefficients,  $\tilde{\Lambda} = \frac{2}{\lambda_{max}} \Lambda - \mathbf{I}_N$ , and  $\lambda_{max}$  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  $\mathbf{g}_\theta$  is:

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

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 [43] 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  $\theta_0$  and  $\theta_1$  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, [44] modeled the traffic network as a general graph rather than treating it as grids, where the monitoring stations in a traffic network represented the nodes in the graph, the connections between stations represented the edges, and the adjacency matrix was 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. [45] 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 [46], 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:

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

where  $\mathbf{X} \in \mathbb{R}^{N \times P}$  is the input,  $P$  represents the number of input features of each node.  $\star_G$  denotes the diffusion convolution,  $k$  is the diffusion step,  $f_\theta$  is a filter and  $\theta \in \mathbb{R}^{K \times 2}$  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 [46] proposes a diffusion convolution layer, defined as:

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

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,  $\sigma$  is the activation function. Based on the diffusion convolution process, [47] designed a new neural network layer that can map the transformation of different dimensional features and extract patterns and features in spatial domain. [48] modified the diffusion process in [46] by utilizing a self-adaptive adjacency matrix, which allowed the model to mine hidden spatial dependency by itself. [49] 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:

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

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

**Attention.** Attention mechanism is first proposed for natural language processing [50], 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

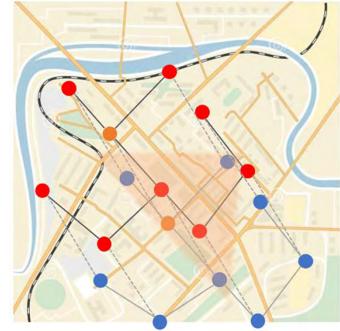


Fig. 4. Spatial-based graph convolution network. Each node in the graph can represent a region in the traffic network. To get a hidden representation of a certain node (e.g. the orange node), GCN aggregates feature information from its neighbors (shaded area). Unlike grid data in 2D images, the neighbors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

correlations between regions in the road network ([51]–[59]). 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:

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

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

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

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, [53] used attention mechanism to dynamically capture the spatial correlation between the target region and the first-order neighboring regions of the road network. [60] 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. Table II shows our classification of recent related literature based on different ways of modeling spatial correlation.

## B. Modeling Temporal Dependency

**CNN.** [97] first introduced the fully convolutional model for sequence to sequence learning. A representative work in

TABLE II  
SPATIAL DEPENDENCY MODELING.

Space division	Category		Approach
Euclidean space	CNN		[1], [61]–[81]
non-Euclidean space	GCN (Spectral-based)	ChebNet	[44], [45], [60], [82]–[90]
		1stChebNet	[44], [74], [91]–[95]
GCN (Spatial-based)			[46]–[49]
GCN+Attention			[60], [89]
Attention only			[51]–[59], [96]

traffic research, [44] 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. [82] and [75] 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 [41]. 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 ([46], [51], [54], [56]–[59], [73], [80], [87], [88], [92], [94], [95], [98], [99]). 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,  $\theta_1$  and  $\theta_2$  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.

[56] designed a temporal attention mechanism to adaptively model the non-linear correlations between different time slices. [60] incorporated a standard convolution and attention mechanism to update the information of a node by fusing the information at the neighboring time slices, and semantically express the dependency intensity between different time slices. Considering that traffic data is highly periodic, but not strictly periodic, [67] designed a periodically shifted attention mechanism to deal with long-term periodic dependency and periodic temporal shifting.

**GCN.** [49] is different from most traffic prediction methods, which capture spatio-temporal relationships using different types of neural network components separately, but uses a component to capture the localized spatio-temporal relationships directly. Specifically, it 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. Table III summarizes relevant literatures in terms of modeling temporal dependency.

### C. Deep Learning plus Traditional Models

Recently, more and more researches are combining deep learning with traditional methods, and some advanced methods have been used in traffic prediction ([106]–[109]). This kind of method not only makes up for the weak ability of non-linear representation of traditional models but also makes up for the poor interpretability of deep learning methods. [106] 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. [107] proposed a deep learning framework that introduced matrix factorization method into deep learning model, which can model the latent

TABLE III  
TEMPORAL DEPENDENCY MODELING.

Category	Approach
CNN	[44], [48], [72], [76], [82], [83], [85], [93]
RNN	[45]–[47], [51]–[53], [61], [62], [64]–[66], [68], [70], [71], [73]–[75], [77], [78], [80], [84], [86], [88]–[91], [94], [95], [98]–[105]
GCN	[49]
CNN+Attention	[60]
RNN+Attention	[59], [67], [87]
Attention only	[54]–[58], [92]

region functions along with the correlations among regions, and further improve the model capability of the citywide flow prediction. [108] 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. [109] combined a latent model and RNN 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.

#### IV. THE STATE-OF-THE-ART RESULTS

In this section, we summarize the state-of-the-art of different application tasks. Table IV shows the classification of recent literatures in related application tasks, which mainly focus on short-to-medium-term prediction. Furthermore, from these papers, we list the current best performance methods under commonly used public datasets, as shown in Table V. We can have the following observations: First, the prediction performance of different algorithms depends heavily on the dataset. More specifically, 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 [110], 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) [111] 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.

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 [112]. 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 [88]. The overall MAPE and 2-hour MAPE of speed prediction on this dataset decreased to 8.63% and 9.78%, respectively.

#### V. PUBLIC DATASETS

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

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

TABLE IV  
LITERATURES FOR DIFFERENT TASKS.

Category	Approach
Flow	[1], [6], [18], [21], [22], [25], [28], [29], [32]–[34], [39], [40], [51], [55], [56], [59], [60], [63], [64], [68], [69], [71], [75], [76], [78]–[80], [82], [85], [90], [94], [96], [107], [109]
Speed	[27], [30], [36], [37], [44], [46]–[48], [51], [53], [55]–[57], [65], [72], [83]–[91], [93], [100], [104], [105]
Demand	[19], [20], [23], [24], [26], [31], [45], [52], [54], [58], [61], [67], [70], [73], [74], [77], [81], [92], [95], [101], [107]
Occupancy	[26], [35], [66], [99], [106], [108]
Travel time	[38], [62], [102], [103]

TABLE V  
STATISTICS PREDICTION FOR DIFFERENT TASKS.

Task	Dataset	Time interval	Prediction window	MAPE	RMSE
Demand	NYC Taxi	30min	30min	–	8.38 [73]
	NYC Bike	60min	60min	21.00% [92]	4.51 [92]
	TaxiBJ	30min	30min	13.80% [92]	17.24 [92]
Speed	METR-LA	5min	5/15/30/60min	4.90% [47]/ 6.80%/8.30%/10.00% [89]	3.57 [47]/ 5.12/6.17/7.30 [89]
	PeMS-BAY	5min	15/30/60min	2.73% [48]/ 3.63% [56]/ 4.31% [56]	2.74 [48]/ 3.70 [48]/ 4.32 [56]
	PeMSD7(M)	5min	15/30/45min	5.24%/7.33%/8.69% [44]	4.04/5.70/6.77 [44]
	SZ-taxi	15min	15/30/45/60min	–	3.92/3.96/3.98/4.00 [91]
	Los-loop	5min	15/30/45/60min	–	5.12/6.05/6.70/7.26 [91]
	LOOP	5min	5min	6.01% [86]	4.63 [86]
	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% [88]	–
Flow	TaxiBJ	30min	30min	–	16.69 [69]
	PeMSD3	5min	60min	16.78% [49]	29.21 [49]
	PeMSD4	5min	60min	11.09% [59]	31.00 [59]
	PEMS07	5min	60min	10.21% [49]	38.58 [46]
	PeMSD8	5min	60min	8.31% [59]	24.74 [59]
	NYC Bike	60min	60min	–	6.33 [69]
	T-Drive	60min	60/120/180min	–	29.9/34.7/37.1 [51]
Travel Time	Chengdu	–	–	11.89% [62]	–
Occupancy	PeMSD-SF	60min	7 rolling time windows (24 time-points at a time)	16.80% [108]	–

- 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/STGCN/tree/master/datasets>.

**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/STGCN>.

**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 entirety 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 Otc. 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/trafficanalysistools/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.

**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.
  - (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 choose NYC Taxi dataset for demand prediction, and METR-LA dataset for speed prediction. We evaluate existing state-of-the-art approaches on these two datasets, and investigate the performance limits.

### A. Experimental Setup

In the experiment, we compare the performance among several typical demand prediction and speed prediction methods with public codes. Next, we will describe these two datasets and the three evaluation criteria used in the experiment in detail.

**NYC Taxi dataset:** The New York taxi trip records used in this experiment contains 22349490 records from 1/1/2015 to 3/1/2015, 60 days in total. We use previous 32 days as training data, 8 days as validation data, and the remaining 20 days as testing data. The traffic data is aggregated every 30 minutes, and Min-max is used to normalize traffic data into [0,1] scale. The whole city is divided into  $10 \times 20$  regions. The size of each region is about  $1km \times 1km$ . Because in the real-world applications, we are more concerned with high traffic, and we filter the demand values less than 10 when testing the models.

**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 [113].

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

$$RMSE = \sqrt{\frac{1}{\xi} \sum_{i=1}^{\xi} (\hat{y}^i - y^i)^2}, \quad (15)$$

$$MAE = \frac{1}{\xi} \sum_{i=1}^{\xi} |\hat{y}^i - y^i|, \quad (16)$$

TABLE VI  
THE PERFORMANCE OF TAXI DEMAND PREDICTION ON NYC TAXI.

Method	Start/Pick-up		End/Drop-off	
	RMSE	MAPE	RMSE	MAPE
ST-ResNet	26.23	21.13%	21.63	21.09%
DMVST-Net	25.71	17.36%	20.50	17.11%
STDN	24.10	16.30%	19.05	16.25%

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

where  $\hat{y}^i$  and  $y^i$  denote the predicted value and the ground truth of region  $i$  for predicted time step, and  $\xi$  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 ([44], [46], [48], [49], [51], [56], [60], [61], [63], [67], [69]).

### B. Experimental Results and Analysis on Demand Prediction

Table VI presents the performance comparison among several advanced traffic demand prediction methods on NYC Taxi dataset, including start/pick-up demand and end/drop-off demand.

ST-ResNet [69] summarized the temporal properties into three categories, consisting of temporal closeness, period, and trend. Then, it used CNN and residual unit to learn spatial dependency on data with different attributes. While it uses traffic information of historical time steps for prediction, it does not explicitly *model the temporal sequential dependency*, leading to lower performance than other comparison methods to a certain extent. DMVST-Net [61] and STDN [67] considered both spatial relation and temporal sequential relation, and used LSTM for modeling temporal sequential dependency. DMVST-Net focused more on local spatial correlations and additionally considered semantic correlations between regions with similar temporal patterns, and STDN used flow information instead of static distance to describe the spatial similarity between regions, and introduced attention mechanisms into LSTM to further track dynamic spatial correlation and dynamic time periodicity. The performance of STDN is nearly 6% better than that of DMVST-Net, and therefore it is crucial to *model the dynamics of spatio-temporal data*.

Although these methods continue to improve performance, regions and their pair-wise relationships are formulated as an Euclidean structure, represented by 2D matrices or images, and consequently convolution neural networks are leveraged for effective prediction. However, the spatial features learned in a CNN are not optimal to represent the traffic network structure.

### C. Experimental Results and Analysis on Speed Prediction

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 are shown in Table VII.

TABLE VII  
THE PERFORMANCE OF TRAFFIC SPEED PREDICTION ON METR-LA.

	$T$	Metric	STGCN	DCRNN	ASTGCN	ST-MetaNet	Graph WaveNet	STSGCN	GMAN
METR-LA	15min	MAE	2.88	2.77	4.86	2.68	2.69	3.01	2.77
		RMSE	5.74	5.38	9.27	5.15	5.15	6.69	5.48
		MAPE	7.62%	7.30%	9.21%	6.89%	6.90%	7.27%	7.25%
	30min	MAE	3.47	3.15	5.43	3.09	3.07	3.42	3.07
		RMSE	7.24	6.45	10.61	6.28	6.22	7.93	6.34
		MAPE	9.57%	8.80%	10.13%	8.45%	8.37%	8.49%	8.35%
	60min	MAE	4.59	3.60	6.51	3.60	3.53	4.09	3.40
		RMSE	9.40	7.59	12.52	7.52	7.37	9.65	7.21
		MAPE	12.70%	10.50%	11.64%	10.46%	10.01%	10.44%	9.72%

STGCN [44] applied ChebNet graph convolution and 1D convolution to extract spatial dependencies and temporal correlations. ASTGCN [60] 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 [46] 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. ST-MetaNet [51] was a deep-meta-learning based model, which used a meta graph attention network to consider diverse spatial correlations, and a meta RNN to capture diverse temporal correlations. Graph WaveNet [48] combined graph convolution with dilated causal convolution to capture spatial-temporal dependencies. STSGCN [49] simultaneously extracted localized spatio-temporal correlation information based on the adjacency matrix of localized spatio-temporal graph. GMAN [56] 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 VII: First, the attention-based methods (ST-MetaNet and GMAN) perform better than other GCN-based methods in extracting spatial correlations. 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. 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 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.

## VII. FUTURE DIRECTION AND OPEN PROBLEMS

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 ([51], [71], [75]), 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. 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. Therefore, how to design an effective lightweight neural network to reduce the amount of network computation and speed up the network is a great challenge.
- **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 ([106]–[109]), 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.

## VIII. CONCLUSION

In this paper, we first summarize the existing traffic prediction methods, and give a taxonomy of them. Then, we list the state-of-the-art in different traffic prediction tasks, and present 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 people 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 *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2016, pp. 1–4.
- [2] E. Vlahogianni, M. Karlaftis, and J. Golias, “Short-term traffic forecasting: Where we are and where were going,” *Transportation Research Part C Emerging Technologies*, vol. 43, no. 1, 2014.
- [3] Y. Li and S. Cyrus, “A brief overview of machine learning methods for short-term traffic forecasting and future directions,” *SIGSPATIAL Special*, vol. 10, no. 1, p. 39, 2018. [Online]. Available: <https://doi.org/10.1145/3231541.3231544>
- [4] A. Singh, A. Shadan, R. Singh, and Ranjeet, “Traffic forecasting,” *International Journal of Scientific Research and Review*, vol. 7, no. 3, 2019.
- [5] P. Xie, T. Li, J. Liu, S. Du, X. Yang, and J. Zhang, “Urban flow prediction from spatiotemporal data using machine learning: A survey,” *Information Fusion*, 2020. [Online]. Available: <https://doi.org/10.1016/j.inffus.2020.01.002>
- [6] B. Williams and L. Hoel, “Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results,” *Journal of transportation engineering*, vol. 129, no. 6, pp. 664–672, 2003.
- [7] E. Zivot and J. Wang, “Vector autoregressive models for multivariate time series,” *Modeling Financial Time Series with S-Plus®*, pp. 385–429, 2006.
- [8] R. Chen, C. Liang, W. Hong, and D. Gu, “Forecasting holiday daily tourist flow based on seasonal support vector regression with adaptive genetic algorithm,” *Applied Soft Computing*, vol. 26, pp. 435–443, 2015.
- [9] U. Johansson, H. Boström, T. Löfström, and H. Linusson, “Regression conformal prediction with random forests,” *Machine Learning*, vol. 97, no. 1–2, pp. 155–176, 2014.
- [10] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Wang, “Traffic flow prediction with big data: A deep learning approach,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 865–873, 2015.
- [11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.
- [12] N. Kalchbrenner and P. Blunsom, “Recurrent continuous translation models,” in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013, pp. 1700–1709.
- [13] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, “Spectral networks and locally connected networks on graphs,” in *Proceedings of International Conference on Learning Representations*, 2014.
- [14] D. Rumelhart, G. Hinton, and R. Williams, “Learning representations by back-propagating errors,” *nature*, vol. 323, no. 6088, pp. 533–536, 1986.
- [15] J. Elman, “Distributed representations, simple recurrent networks, and grammatical structure,” *Machine learning*, vol. 7, no. 2–3, pp. 195–225, 1991.
- [16] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [17] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder-decoder for statistical machine translation,” *arXiv preprint arXiv:1406.1078*, 2014.
- [18] S. Shekhar and B. Williams, “Adaptive seasonal time series models for forecasting short-term traffic flow,” *Transportation Research Record*, vol. 2024, no. 1, pp. 116–125, 2007.
- [19] 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 of Computer Science*, vol. 6, no. 1, pp. 111–121, 2012.
- [20] L. Moreira-Matias, J. Gama, M. Ferreira, J. Mendes-Moreira, and L. Damas, “Predicting taxi-passenger demand using streaming data,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1393–1402, 2013.
- [21] M. Lippi, M. Bertini, and P. Frasconi, “Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 871–882, 2013.
- [22] I. Wagner-Muns, I. Guardiola, V. Samaranayke, and W. Kayani, “A functional data analysis approach to traffic volume forecasting,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 878–888, 2017.

- [23] W. Li, J. Cao, J. Guan, S. Zhou, G. Liang, W. So, and M. Szczecinski, "A general framework for unmet demand prediction in on-demand transport services," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 8, pp. 2820–2830, 2018.
- [24] 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. 32 005–32 014, 2018.
- [25] Z. Diao, D. Zhang, X. Wang, K. Xie, S. He, X. Lu, and Y. Li, "A hybrid model for short-term traffic volume prediction in massive transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 3, pp. 935–946, 2018.
- [26] D. Salinas, M. Bohlke-Schneider, L. Callot, R. Medico, and J. Gasthaus, "High-dimensional multivariate forecasting with low-rank gaussian copula processes," in *Advances in Neural Information Processing Systems*, 2019, pp. 6824–6834.
- [27] L. Lin, J. Li, F. Chen, J. Ye, and J. Huai, "Road traffic speed prediction: a probabilistic model fusing multi-source data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 7, pp. 1310–1323, 2017.
- [28] P. Duan, G. Mao, W. Liang, and D. Zhang, "A unified spatio-temporal model for short-term traffic flow prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 9, pp. 3212–3223, 2018.
- [29] H. Tan, Y. Wu, B. Shen, P. Jin, and B. Ran, "Short-term traffic prediction based on dynamic tensor completion," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 8, pp. 2123–2133, 2016.
- [30] J. Shin and M. Sunwoo, "Vehicle speed prediction using a markov chain with speed constraints," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 9, pp. 3201–3211, 2018.
- [31] K. Ishibashi, S. Harada, and R. Kawahara, "Inferring latent traffic demand offered to an overloaded link with modeling qos-degradation effect," *IEICE Transactions on Communications*, 2018.
- [32] 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 *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 2018, pp. 1243–1252.
- [33] N. Polson and V. Sokolov, "Bayesian particle tracking of traffic flows," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 2, pp. 345–356, 2017.
- [34] H. Hong, X. Zhou, W. Huang, X. Xing, F. Chen, Y. Lei, K. Bian, and K. Xie, "Learning common metrics for homogenous tasks in traffic flow prediction," in *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2015, pp. 1007–1012.
- [35] H. Yu, N. Rao, and I. Dhillon, "Temporal regularized matrix factorization for high-dimensional time series prediction," in *Advances in neural information processing systems*, 2016, pp. 847–855.
- [36] 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 *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1525–1534.
- [37] D. Deng, C. Shahabi, U. Demiryurek, and L. Zhu, "Situation aware multi-task learning for traffic prediction," in *2017 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2017, pp. 81–90.
- [38] A. Kinoshita, A. Takasu, and J. Adachi, "Latent variable model for weather-aware traffic state analysis," in *International Workshop on Information Search, Integration, and Personalization*. Springer, 2016, pp. 51–65.
- [39] Y. Gong, Z. Li, J. Zhang, W. Liu, and J. Yi, "Potential passenger flow prediction: A novel study for urban transportation development," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.
- [40] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.
- [41] 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, 2018.
- [42] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Advances in neural information processing systems*, 2016, pp. 3844–3852.
- [43] T. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proceedings of the International Conference on Learning Representations*, 2017.
- [44] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018, pp. 3634–3640.
- [45] X. Geng, Y. Li, L. Wang, L. Zhang, Q. Yang, J. Ye, and Y. Liu, "Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 3656–3663.
- [46] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural networks: data-driven traffic forecasting," in *Proceedings of the International Conference on Learning Representations*, 2018.
- [47] C. Chen, K. Li, S. Teo, X. Zou, K. Wang, J. Wang, and Z. Zeng, "Gated residual recurrent graph neural networks for traffic prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 485–492.
- [48] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph wavenet for deep spatial-temporal graph modeling," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. AAAI Press, 2019, pp. 1907–1913.
- [49] C. Song, Y. Lin, S. Guo, and H. Wan, "Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting," <https://github.com/wanhuaizy/STGCN/blob/master/paper/AAAI2020-STGCN.pdf>, 2020.
- [50] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.
- [51] 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 *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 1720–1730.
- [52] Y. Li, Z. Zhu, D. Kong, M. Xu, and Y. Zhao, "Learning heterogeneous spatial-temporal representation for bike-sharing demand prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 1004–1011.
- [53] J. Zhang, X. Shi, J. Xie, H. Ma, I. King, and D. Yeung, "Gaan: Gated attention networks for learning on large and spatiotemporal graphs," *arXiv preprint arXiv:1803.07294*, 2018.
- [54] Y. Li and J. Moura, "Forecaster: A graph transformer for forecasting spatial and time-dependent data," *arXiv preprint arXiv:1909.04019*, 2019.
- [55] X. Yi, Z. Duan, T. Li, T. Li, J. Zhang, and Y. Zheng, "Citytraffic: Modeling citywide traffic via neural memorization and generalization approach," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2019, pp. 2665–2671.
- [56] C. Zheng, X. Fan, C. Wang, and J. Qi, "Gman: A graph multi-attention network for traffic prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.
- [57] C. Park, C. Lee, H. Bahng, K. Kim, S. Jin, S. Ko, and J. Choo, "Stgrat: A spatio-temporal graph attention network for traffic forecasting," *arXiv preprint arXiv:1911.13181*, 2019.
- [58] X. Geng, L. Zhang, S. Li, Y. Zhang, L. Zhang, L. Wang, Q. Yang, H. Zhu, and J. Ye, "{CGT}: Clustered graph transformer for urban spatio-temporal prediction," in *Proceedings of the International Conference on Learning Representations*, 2020. [Online]. Available: <https://openreview.net/forum?id=H1eJAANtv>
- [59] X. Shi, H. Qi, Y. Shen, G. Wu, and B. Yin, "A spatial-temporal attention approach for traffic prediction," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2020.
- [60] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 922–929.
- [61] H. Yao, F. Wu, J. Ke, X. Tang, Y. Jia, S. Lu, P. Gong, J. Ye, and Z. Li, "Deep multi-view spatial-temporal network for taxi demand prediction," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [62] D. Wang, J. Zhang, W. Cao, J. Li, and Y. Zheng, "When will you arrive? estimating travel time based on deep neural networks," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [63] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 1020–1027.
- [64] P. Zhao, D. Cai, S. Zhang, F. Chen, Z. Zhang, C. Wang, and J. Li, "Layerwise recurrent autoencoder for general real-world traffic flow forecasting," 2018.
- [65] Z. Lv, J. Xu, K. Zheng, H. Yin, P. Zhao, and X. Zhou, "Lc-rnn: a deep learning model for traffic speed prediction," in *Proceedings of the*

- 27th International Joint Conference on Artificial Intelligence*, 2018, pp. 3470–3476.
- [66] G. Lai, W. Chang, Y. Yang, and H. Liu, “Modeling long-and short-term temporal patterns with deep neural networks,” in *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 2018, pp. 95–104.
- [67] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li, “Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 5668–5675.
- [68] A. Zonoozi, J. Kim, X. Li, and G. Cong, “Periodic-crn: a convolutional recurrent model for crowd density prediction with recurring periodic patterns,” in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018, pp. 3732–3738.
- [69] J. Zhang, Y. Zheng, and D. Qi, “Deep spatio-temporal residual networks for citywide crowd flows prediction,” in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.
- [70] J. Ke, H. Zheng, H. Yang, and X. Chen, “Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach,” *Transportation Research Part C: Emerging Technologies*, vol. 85, pp. 591–608, 2017.
- [71] L. Wang, X. Geng, X. Ma, F. Liu, and Q. Yang, “Cross-city transfer learning for deep spatio-temporal prediction,” *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018.
- [72] 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 Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3700–3709, 2018.
- [73] 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 *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2019, pp. 305–313.
- [74] N. Davis, G. Raina, and K. Jagannathan, “Grids versus graphs: Partitioning space for improved taxi demand-supply forecasts,” *arXiv preprint arXiv:1902.06515*, 2019.
- [75] H. Yao, Y. Liu, Y. Wei, X. Tang, and Z. Li, “Learning from multiple cities: A meta-learning approach for spatial-temporal prediction,” in *The World Wide Web Conference*. ACM, 2019, pp. 2181–2191.
- [76] S. Guo, Y. Lin, S. Li, Z. Chen, and H. Wan, “Deep spatial-temporal 3d convolutional neural networks for traffic data forecasting,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3913–3926, 2019.
- [77] L. Liu, Z. Qiu, G. Li, Q. Wang, W. Ouyang, and L. Lin, “Contextualized spatial-temporal network for taxi origin-destination demand prediction,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3875–3887, 2019.
- [78] Z. Zheng, Y. Yang, J. Liu, H. Dai, and Y. Zhang, “Deep and embedded learning approach for traffic flow prediction in urban informatics,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3927–3939, 2019.
- [79] J. Zhang, Y. Zheng, J. Sun, and D. Qi, “Flow prediction in spatio-temporal networks based on multitask deep learning,” *IEEE Transactions on Knowledge and Data Engineering*, 2019.
- [80] R. Jiang, X. Song, D. Huang, X. Song, T. Xia, Z. Cai, Z. Wang, K. Kim, and R. Shibasaki, “Deepurbanevent: A system for predicting citywide crowd dynamics at big events,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2019, pp. 2114–2122.
- [81] D. Lee, S. Jung, Y. Cheon, D. Kim, and S. You, “Forecasting taxi demands with fully convolutional networks and temporal guided embedding,” in *In Advances in Neural Information Processing Systems*, 2018.
- [82] S. Fang, Q. Zhang, G. Meng, S. Xiang, and C. Pan, “Gstnet: Global spatial-temporal network for traffic flow prediction,” in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI*, 2019, pp. 10–16.
- [83] Z. Diao, X. Wang, D. Zhang, Y. Liu, K. Xie, and S. He, “Dynamic spatial-temporal graph convolutional neural networks for traffic forecasting,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 890–897.
- [84] 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,” *arXiv preprint arXiv:1904.06656*, 2019.
- [85] M. Wang, B. Lai, Z. Jin, Y. Lin, X. Gong, J. Huang, and X. Hua, “Dynamic spatio-temporal graph-based cnns for traffic prediction,” *arXiv preprint arXiv:1812.02019*, 2018.
- [86] 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 Transactions on Intelligent Transportation Systems*, 2019.
- [87] 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,” *arXiv preprint arXiv:1810.10237*, 2018.
- [88] B. Liao, J. Zhang, C. Wu, D. McIlwraith, T. Chen, S. Yang, Y. Guo, and F. Wu, “Deep sequence learning with auxiliary information for traffic prediction,” in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 537–546.
- [89] W. Chen, L. Chen, Y. Xie, W. Cao, Y. Gao, and X. Feng, “Multi-range attentive bicomponent graph convolutional network for traffic forecasting,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.
- [90] K. Guo, Y. Hu, Z. Qian, H. Liu, K. Zhang, Y. Sun, J. Gao, and B. Yin, “Optimized graph convolution recurrent neural network for traffic prediction,” *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–12, 2020.
- [91] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, “T-gcn: A temporal graph convolutional network for traffic prediction,” *IEEE Transactions on Intelligent Transportation Systems*, 2019.
- [92] L. Bai, L. Yao, S. Kanhere, X. Wang, and Q. Sheng, “Stg2seq: spatial-temporal graph to sequence model for multi-step passenger demand forecasting,” in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. AAAI Press, 2019, pp. 1981–1987.
- [93] K. Lee and W. Rhee, “Graph convolutional modules for traffic forecasting,” *arXiv preprint arXiv:1905.12256*, 2019.
- [94] D. Chai, L. Wang, and Q. Yang, “Bike flow prediction with multi-graph convolutional networks,” in *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2018, pp. 397–400.
- [95] 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 *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2019, pp. 1227–1235.
- [96] Z. Liu, F. Miranda, W. Xiong, J. Yang, Q. Wang, and C. T.Silva, “Learning geo-contextual embeddings for commuting flow prediction,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.
- [97] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. Dauphin, “Convolutional sequence to sequence learning,” in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR.org, 2017, pp. 1243–1252.
- [98] L. Zhu and N. Laptev, “Deep and confident prediction for time series at uber,” in *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2017, pp. 103–110.
- [99] P. Deshpande and S. Sarawagi, “Streaming adaptation of deep forecasting models using adaptive recurrent units,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 1560–1568.
- [100] X. Wang, C. Chen, Y. Min, J. He, B. Yang, and Y. Zhang, “Efficient metropolitan traffic prediction based on graph recurrent neural network,” *arXiv preprint arXiv:1811.00740*, 2018.
- [101] J. Xu, R. Rahmatizadeh, L. Bölöni, and D. Turgut, “Real-time prediction of taxi demand using recurrent neural networks,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 8, pp. 2572–2581, 2017.
- [102] 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 Transactions on Intelligent Transportation Systems*, vol. 20, no. 9, pp. 3283–3293, 2018.
- [103] P. He, G. Jiang, S. Lam, and D. Tang, “Travel-time prediction of bus journey with multiple bus trips,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 11, pp. 4192–4205, 2018.
- [104] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.
- [105] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, “Long short-term memory neural network for traffic speed prediction using remote

- microwave sensor data,” *Transportation Research Part C: Emerging Technologies*, vol. 54, pp. 187–197, 2015.
- [106] L. Li, J. Yan, X. Yang, and Y. Jin, “Learning interpretable deep state space model for probabilistic time series forecasting,” in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 2019, pp. 2901–2908.
- [107] 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 *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2019, pp. 2683–2691.
- [108] R. Sen, H. Yu, and I. Dhillon, “Think globally, act locally: A deep neural network approach to high-dimensional time series forecasting,” in *Advances in Neural Information Processing Systems*, 2019, pp. 4838–4847.
- [109] A. Ziat, E. Delasalles, L. Denoyer, and P. Gallinari, “Spatio-temporal neural networks for space-time series forecasting and relations discovery,” in *2017 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2017, pp. 705–714.
- [110] amap technology, “amap technology annual in 2019,” <https://files.alicdn.com/tpsservice/46a2ae997f5ef395a78c9ab751b6d942.pdf>.
- [111] C. Lea, M. Flynn, R. Vidal, A. Reiter, and G. Hager, “Temporal convolutional networks for action segmentation and detection,” in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 156–165.
- [112] X. Zhang, L. Xie, Z. Wang, and J. Zhou, “Boosted trajectory calibration for traffic state estimation,” in *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019, pp. 866–875.
- [113] D. Shuman, S. 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 processing magazine*, vol. 30, no. 3, pp. 83–98, 2013.