

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Prediction of Large Scale Spatio-temporal Traffic Flow Data with New Graph Convolution Model

Ping Wang, Tongtong Shi, Rui He and Wubei Yuan

Abstract

Prompt and accurate prediction of traffic flow is quite useful. It will help traffic administrator to analyze the road occupancy status and formulate dynamic and flexible traffic control in advance to improve the road capacity. It can also provide more precise navigation guidance for the road users in future. However, it is hard to predict spatio-temporal traffic flow data in large scale promptly with high accuracy caused by complex interrelation and nonlinear dynamic nature. With development of deep learning and other technologies, many prediction networks could predict traffic flow with accumulated historical data in time series. In consideration of the regional characteristics of traffic flow, the emerging Graph Convolutional Network (GCN) model is systematically introduced with representative applications. Those successful applications provide a possible way to contribute fast and proper traffic control strategies that could relieve traffic pressure, reduce potential conflict, fasten emergency response, etc.

Keywords: traffic flow, GCN, traffic data, deep learning, ITS

1. Introduction

1.1 Background and current status

Traffic problems such as frequent traffic congestion, serious traffic accidents, and long commuting times have seriously reduced the travel experience of passengers and the efficiency of traffic operations [1]. To cope with these problems, researchers work on improving the traffic control strategies based on prediction of future traffic status [2]. Traffic flow is one of important road conditions to access [3]. Based on prompt and accuracy perdition, better and fast-adjusted traffic control and guidance could be applied. Therefore, reliable traffic flow prediction is also one of the key factors to upgrade the traffic system from “passive adjust” to “active control in advance”; even prediction of future short-term traffic status of road sections is quite useful to prevent congestion deteriorate. For traffic management departments, early detection of traffic instability and abnormal potential risks based on reliable prediction data can improve a large number of existing traffic management control applications, such as traffic calming, signal control, etc.; for road users, real-time route updates and adjustments based on dynamic traffic prediction results can adjust travel time and

routes before congestion develops, thus providing vehicles to plan a driving path to avoid congested road sections and congested intersections or to plan a path with the shortest driving time for vehicles to improve traffic efficiency.

The current traffic prediction also faces the following challenges, as shown in **Figure 1**: (1) analyzing the spatial correlation of the road network: some roads are adjacent to each other and have different degrees of influence on upstream and downstream traffic volumes, so the traffic flow in this part is spatially correlated, and it is a challenge to consider the spatial location relationship to correlate the neighboring traffic flow characteristics [4]; (2) unlike the regular network layout, the traffic map structure is irregular; (3) the nonlinear retention of medium and long time prediction models: the traffic flow changes drastically at the peak time, which is difficult to predict, especially as the prediction time increases, the nonlinear retention ability of the model decreases and the time series signal gradually decays, so how to better correlate the time series relationship of traffic flow to maintain the steady-state time series prediction is also a long-term challenging task [5].

1.2 Related works

Traffic flow forecasting is based on historical traffic flow data to predict future traffic flow, which is a typical regression problem of traffic network time series [6]. In order to solve the traffic flow forecasting problem, factors such as traffic patterns, data types, spatial locations, and time periods need to be considered. Nowadays, many computational forecasting methods have been widely used in traffic flow prediction and have achieved good research results. As shown in **Figure 2**, common traffic flow prediction methods can be divided into three major categories [7]. ① early traffic flow prediction methods; ② machine-learning-based traffic flow prediction methods; ③ deep-learning-based traffic flow prediction methods.

1.2.1 Early traffic flow prediction methods

Early traffic flow prediction methods mainly model the relationship between traffic flow, speed, and density and regress the traffic flow data as well as optimize the parameters to achieve the fitting prediction of traffic data, mainly including statistical models and traffic simulation.

1.2.1.1 Typical methods

- Miska et al. [8] proposed cellular automata (CA) to simulate each participant of different flows and their interaction phenomena.

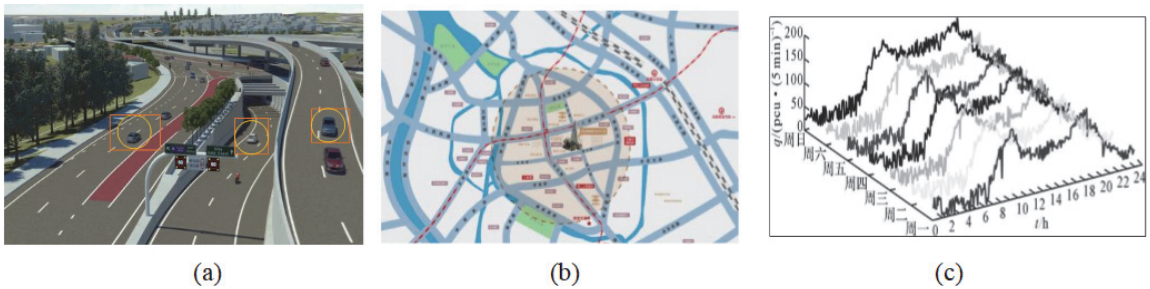


Figure 1. Challenges in traffic flow prediction; (a) road relevance from the web (https://www.ivsky.com/tupian/daolu_t948/); (b) the complex road network map of Shaanxi Province; (c) periodicity of traffic data.

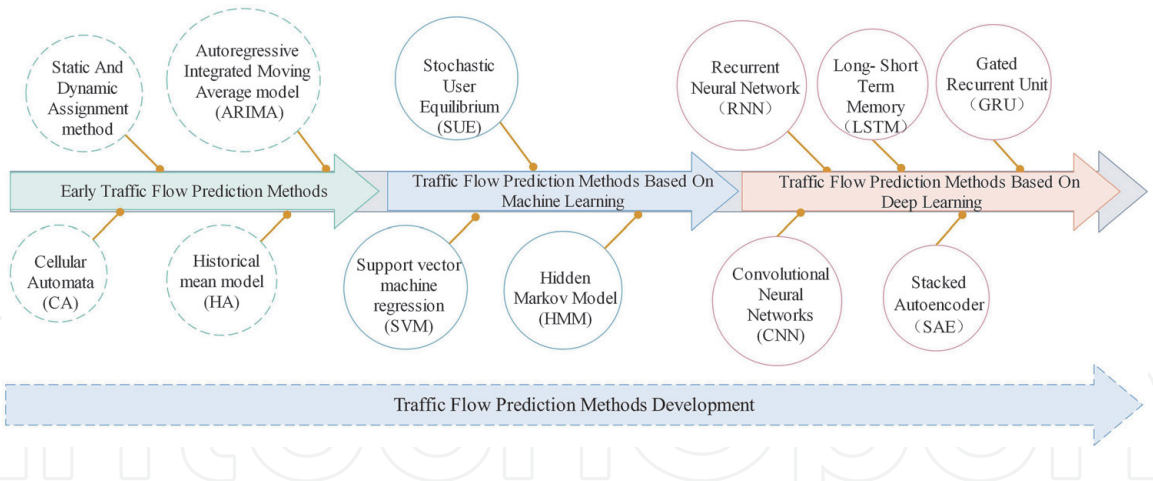


Figure 2.
Classification of time-line based traffic flow prediction methods.

- Ngoduy et al. [9] proposed the use of static and dynamic assignment methods to allocate traffic on a simulated road network.
- Stephanedes et al. [10] have applied the historical mean model (HA) in urban traffic control systems in 1984.
- Kumer et al. [11] used the Autoregressive Integrated Moving Average (ARIMA) model to represent the predicted traffic flow in the form of a mathematical model.

1.2.1.2 Advantages and disadvantages

Models such as statistical mathematical models and traffic simulations can describe this traffic flow prediction as a time series problem approximately. However, simulation systems and simulation tools still need to consume a lot of computational power and skilled parameter settings to reach a steady state, and it is more difficult to get accurate prediction results from this prediction model due to the complexity of traffic scenarios. Besides, these methods based on statistics are only applicable to linear data, while traffic flow data are nonlinear and complex; thus, such methods are not capable of handling complex nonlinear traffic data.

1.2.2 Traffic flow prediction methods based on machine learning

With the demand for high accuracy in intelligent traffic scenarios, the shortcomings of traditional prediction methods that cannot model the complex state of traffic flow become more and more prominent, and machine learning methods gradually take an important place in traffic flow prediction tasks.

1.2.2.1 Typical methods

- YS et al. [12] proposed a traffic flow prediction method using support vector machine regression. The idea of using support vector machine method is to map the low-dimensional nonlinear traffic data to a high-dimensional space by introducing a kernel function before linear classification.
- Zhu et al. [13] predict the path traffic volume and roadway flow by building a Bayesian network model.

- Qi et al. [14] proposed a Hidden Markov Model (HMM) for short-term highway traffic prediction.

1.2.2.2 Advantages and disadvantages

Machine learning methods can better model the stochastic processes and nonlinear properties of traffic flows and have mostly better performance compared with traditional models. However, such methods do not consider the spatial and temporal correlation of traffic flow data and require extensive feature engineering. Therefore, it is difficult to solve complex traffic flow prediction problems.

1.2.3 Traffic flow prediction methods based on deep learning

Deep learning has been very successful in the fields of computer vision, speech recognition, and natural language processing, and more and more scholars are applying deep neural networks (DNNs) to various real-world scenario tasks. In traffic flow prediction, the models can be classified into road section prediction and area prediction according to their prediction range characteristics.

1.2.3.1 Typical methods

- Chen et al. [15] proposed a convolutional neural network (CNN)-based traffic flow prediction method using time series folding for multi-scale learning.
- Lv et al. [16] proposed a heap-based autoencoder (SAE) method for traffic flow prediction considering spatiotemporal relationships.
- Yu et al. [17] proposed a Long-Short Term Memory (LSTM)-based method for traffic flow prediction on road networks during peak periods.
- Cho K et al. [18] proposed a gated recurrent unit that can establish links for traffic data at adjacent moments and preserve the memory by gating and other means to learn long-term dependencies of traffic flow sequences.
- Yu et al. [19] proposed a spatiotemporal graph convolutional network (STGCN) for traffic flow velocity prediction on a multi-scale traffic network.
- The ST-ResNet [20] model restricts the input to grid data rather than graph structure in the traffic prediction problem, which makes it difficult to make predictions on complex highway data.
- Geom-GCN [21] proposed a network to update the node representation, but it could not capture the distance dependence between nodes.
- The DCRNN model (2018ICLR) [22] models spatial correlation as a diffusion process on directed graphs to model the translation of traffic flows and proposes diffusion convolutional recurrent neural networks capable of capturing spatial and temporal dependencies between time series using the seq2seq framework.
- The GMAN model (2020AAAI) [23] uses an attention mechanism to model dynamic spatial and nonlinear temporal relationships, respectively.

ASTGCN [24] considers only low-order neighborhood relationships between nodes and ignores correlations between different historical time periods.

1.2.3.2 Advantages and disadvantages

The core of the traffic prediction problem lies in how to effectively capture the spatiotemporal dimensional features and correlations of the data. Traditional convolutional neural networks can effectively extract local features of data, but can only work on standard grid data. The graph convolution can directly extract features from graph structured data and automatically mine the spatial patterns of traffic data. The convolution operation along the time axis can extract the temporal patterns of traffic data. Therefore, this paper focuses on the deep learning model based on graph convolutional network to capture the spatial and temporal characteristics of traffic data and effectively solve the traffic flow prediction problem.

1.2.4 Future directions for exploring traffic flow prediction models

GCN has become a mainstream method in the field of traffic flow prediction, but it started late, and its theoretical foundation and research depth are far from enough. At present, it still faces many problems that need to be solved. There are three main directions as follows.

1. Dynamic graph modeling: Most graph structures processed by GCN networks are static graphs, and there are fewer models involving dynamic graph structures. The graph structure of static graphs is static and unchanging, while the vertices and edges of dynamic graphs change randomly or even disappear, making it difficult to follow the rules.
2. Heterogeneous graph modeling: Homogeneous graph means that nodes and edges are only one type, and this kind of data is easier to handle. The heterogeneous graph refers to the type of nodes and edges, the same node and different node connections will show different properties, the same edge and different node connections will also show different relationships, heterogeneous graph structure is relatively complex to deal with. However, the heterogeneous graph is the most relevant scenario to the actual problem.
3. Deepening the model structure: One of the inherent advantages of GCN is that it smoothes the graph signal, but as its layers keep deepening, its training results are highly susceptible to over smoothing. Since graph convolution is a special form of Laplacian smoothing, the smoothing operation makes the signal more consistent at the feature level as the graph convolution aggregates the features of neighboring nodes, thereby causing the signal to lose its diversity and leading to a sharp performance degradation in the relevant prediction task, a phenomenon that is more pronounced on small data sets. Therefore, GCNs cannot be stacked continuously and deeply like general convolutional models, but shallow neural networks suffer from limited perceptual field and feature extraction capabilities.

2. Traffic prediction based on graph convolution

This section introduces the principles and techniques related to traffic flow prediction based on graph neural networks. First, an overview of graph neural

networks is given, and the graph convolutional networks (GCNs) [25] used to capture the spatial dependence of traffic flows in the road network are introduced separately in this paper. Secondly, the transformation of graph structure into actual road traffic graph structure modeling method is introduced; finally, this paper models the GCN on urban road networks and uses the topology of the GCN capture graph to handle the spatiotemporal traffic prediction task, and the application scenarios of traffic flow prediction are added at the end of the paper.

2.1 Basic graph theory and convolutional networks

2.1.1 Graph theory

Graph is a common data structure that is an important object of study in the field of computer and data science [26]. A graph usually consists of two elements, Vertex and Edge, where the vertices correspond to an abstract representation of the object of study and the edges represent the interconnection between two of the objects. Graphs are often used to represent things and specific relationships between things; in fact, graphs can represent any system with binary relationships. Graphs have a very wide range of applications in real life; social networks of human life, citation systems, urban transportation networks, and biochemical molecules can be effectively represented by graph structures.

2.1.1.1 Basic concept

In graph theory, a graph is usually represented as a set of vertices and edges [27], denoted as $G = (V, E)$, where $V = \{v_1, v_2, \dots, v_n\}$ is the set of vertices and the elements in this non-empty set are the vertices of the graph. The set of edges can be denoted as E . A graph can be classified into directed and undirected graphs depending on whether the edges in the graph have directionality or not. If the edges of a graph have directionality, such edges are called directed edges as in **Figure 3(a)**, and if the edges of a graph have no directionality, then the corresponding graph is an undirected graph as in **Figure 3(b)**. The graphs can be classified into weighted and unweighted graphs according to the presence or absence of specific weights of the edges in the graph [28]. Each edge in a weighted graph has a real weight as in **Figure 3(c)**, which represents the degree of connection between two vertices or the “distance” between two vertices. For example, in a traffic network, the weight of an edge can characterizes the physical distance between two vertices. In contrast, the other category is the unweighted graph, which can also be understood as the unweighted graph in which all the edge weights are equal.

2.1.1.2 Algebraic representation of graphs

As a common data structure, graphs have many kinds of algebraic representations, and common storage representations include adjacency matrices [29],

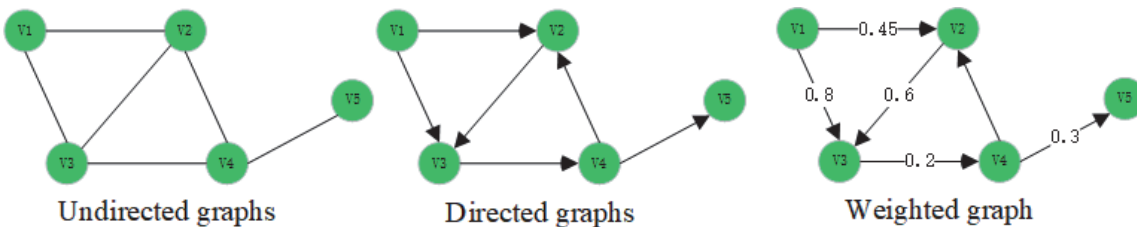


Figure 3.
Basic types of common diagrams.

adjacency tables, and association matrices. Among them, adjacency matrices are widely used in graph representation learning because they can represent the constructional properties of graphs well and are easy to combine with matrix operations to understand the structural features of graphs.

If two vertices of an edge in a graph are v_i and v_j , then v_i and v_j are said to be their respective neighbors. We define the set of neighbors of v_i as $N(v_i)$:

$$N(v_i) = \{v_j | \exists e_{ij} \in E \text{ or } e_{ji} \in E\} \quad (1)$$

The degree of v_i is defined as: the number of edges with v_i as the endpoint, denoted as $\deg(v_i)$, and therefore, $\deg(v_i) = N(v_i)$. In a directed graph, the degree of a vertex can be divided into out-degree and in-degree. The number of directed edges starting at vertex v_i is called the out-degree of v_i , and the number of directed edges ending at vertex v_i is called the in-degree of v_i . The sum of the entry and exit degrees of a vertex is equal to the degree of the vertex, and the sum of the degrees of all nodes is equal to twice the number of all edges. e_{ij} and e_{ji} represent edges in different directions between two vertices, e.g., e_{ij} represents an edge in the direction from i to j , while e_{ji} is the opposite.

The degree matrix is a matrix of the degrees of the vertices, so that the elements at the main diagonal positions are the vertex degrees and the remaining elements are 0. Accordingly, the directed graph has an entry degree matrix and an exit degree matrix. The adjacency matrix is a matrix used to represent the relationship between vertices. For graph $G = (V, E)$, the adjacency matrix can be expressed as:

$$A_{ij} = \begin{cases} 1 & \text{if } \langle v_i, v_j \rangle \subseteq E \\ 0 & \text{else} \end{cases} \quad (2)$$

The core idea of the adjacency table of a graph is to have a neighbor table for each vertex of the vertex set. The association matrix is used to represent the direct association of nodes and edges and is defined as:

$$B_{ij} = \begin{cases} 1 & \text{if } v_i \text{ and } v_j \text{ are connected} \\ 0 & \text{else} \end{cases} \quad (3)$$

The Laplace matrix [30] is a special matrix that is often used in graph theory to study the structural properties of graphs. The Laplace matrix is defined as $L = D - A$, where D is the degree matrix of the graph and A is the adjacency matrix of the graph. **Figure 4** shows the Laplace matrix representation of a simple graph.

2.1.2 Graph convolutional networks

Previous classical convolutional networks based on deep learning mostly consider regular data in Euclidean space in processing data. When inputting ordered data with fixed dimensions (e.g., images, speech, video, etc.), the convolutional operation and the capture and compression of the pooling layer make the network fitting effect remarkable. However, when faced with sequentially disordered road network traffic data with variable dimensions, the suitability of the traditional convolution operation decreases. However, graph neural networks (GNNs) can handle the abovementioned irregular graphs by passing node features into the neural network during iteration and outputting the node states. The original GNNs converge the hidden state to a fixed point based on the “immobile point” theory, which is ineffective for extracting edge information, and in the specific scenario

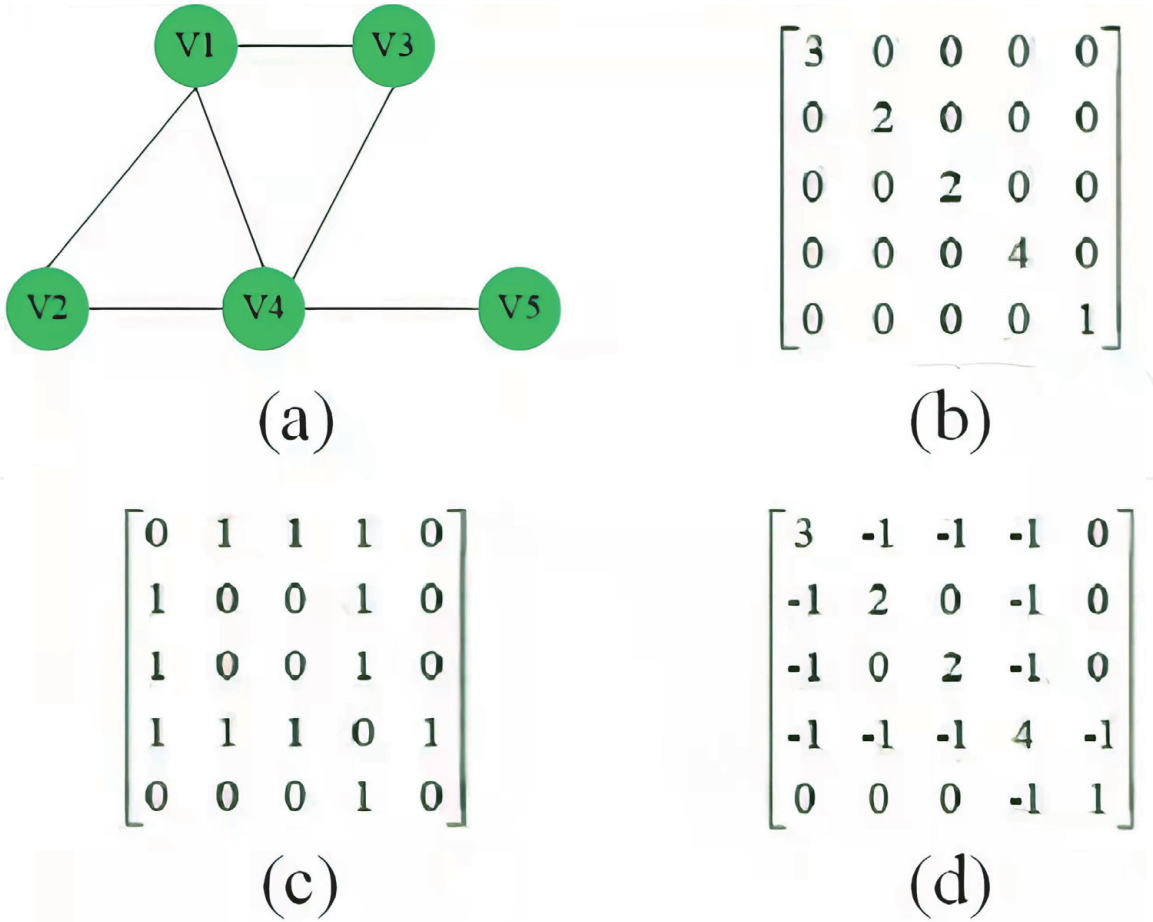


Figure 4. Matrix representation of the graph; (a) graph structure; (b) degree matrix; (c) adjacency matrix; (d) Laplace matrix.

represented by the graph, some feature information is shared among nodes due to the fixed convergence, making the actual information obtained scarce. Therefore, two types of Graph Convolutional Network (GCN) based on frequency domain and null domain are generated. Two types of GCN models: null domain convolution is the same as the traditional convolution method, which can convolve directly at the pixel point of the picture; frequency domain convolution needs to start from the graph signal processing, treating the kernel in the convolution as a filter and the learned features as signals for weighted summation.

As shown in **Figure 5**, the common network framework for graph convolution is illustrated. First, the neighboring nodes of the input graph structure are updated with a layer of convolution operation, and then a layer of ReLU activation function is added to obtain the basic convolution layer plus activation function structure. The above structure is stacked sequentially until the number of stacked layers reaches the prediction of the model, and the output part transforms the node features into labels for the relevant tasks. Unlike GNN circular iterative parameter sharing, GCN is a multilayer stack and the parameters are different for each layer.

Further, it mainly includes graph convolution based on the spectral domain (frequency domain) and the null domain. The spectral domain approach is to construct CNN simulations into the spectral domain by considering the localization of graph convolution through spectral analysis, such as Spectral Graph Convolution (SGC), which mainly focuses on the continuous derivation and improvement of the core formulations of spectral graph theory to reduce the computational power of the model from the perspective of optimization parameters. Empty domain

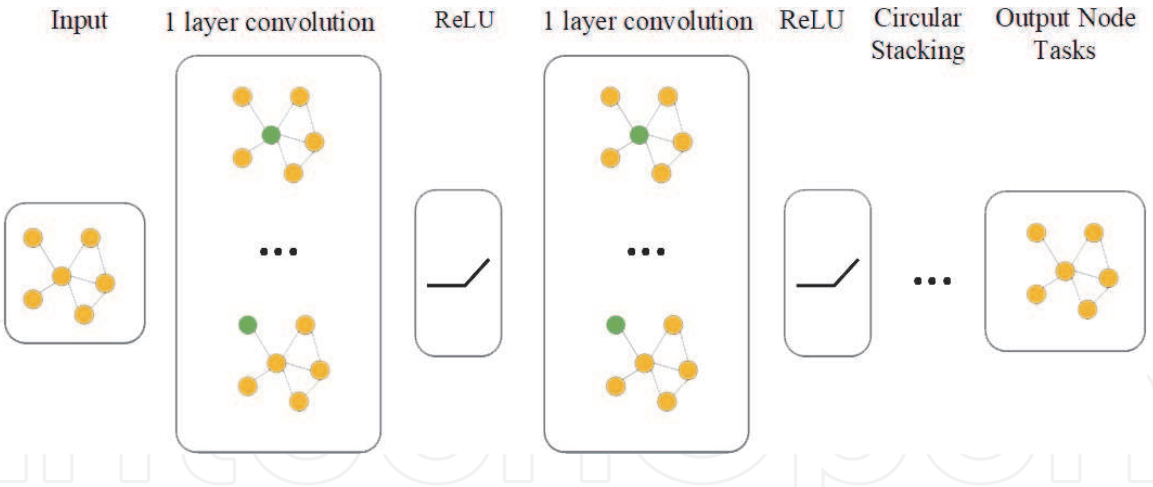


Figure 5.
General framework of graph convolution.

methods perform convolution filters directly on the nodes of the graph and their neighborhoods, such as Diffusion Graph Convolution (DGC).

2.1.2.1 Spectral Domain-based Graph Convolution Network (SGC)

Spectral domain approach [31]: The absence of graph translation invariance poses difficulties in defining convolutional neural networks in the nodal domain. The spectral domain approach uses the convolution theorem to define the graph convolution from the spectral domain. The spectral domain graph convolution network is proposed based on graph signal processing, where the convolution layer of the graph neural network is defined as a filter, i.e., the filter removes the noise signal to obtain the result of the input signal. In practical applications, it can only be used to process graph structures that are undirected and have no information on the edges. The Fourier transform of the signal $f(x)$ and its inverse transform are:

$$F(w) = \varphi(w) = \int_{-\infty}^{+\infty} f(x) \exp(-iwx) dx \tag{4}$$

$$f(x) = \varphi^{-1}(F(w)) \tag{5}$$

where φ denotes the Fourier transform. It can be found that the Fourier transform that changes the time domain to the spectral domain is essentially the integral of the summation $f(x)$ with $\exp(-iwx)$ as the basis vector. Defining the graph G of the input signal as a characteristic decomposable Laplace matrix $L = D - A$, the normalized Laplace matrix L is defined as:

$$L = I_n - D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \tag{6}$$

where D denotes the degree matrix of graph G , A denotes the adjacency matrix, and I_n is the unit matrix of order n . After performing the eigendecomposition, it can be expressed as the universal structure $L = U \Lambda U^T$, where Λ is a matrix with each eigenvalue as a diagonal element, and U is a vector matrix composed of eigenvectors corresponding to each eigenvalue. Since U is an orthogonal matrix, the basis of the conventional Fourier transform $\exp(-iwx)$ is then replaced by U^T and expressed in matrix form to obtain the Fourier transform of the signal x on the graph as:

$$\hat{x} = U^T x \tag{7}$$

where x refers to the original representation of the signal, \hat{x} refers to the signal x after transforming it to the spectral domain, and U^T denotes the transpose of the eigenvector matrix for doing the Fourier transform. The inverse Fourier transform of the signal x is:

$$x = U\hat{x} \quad (8)$$

Using the Fourier transform and the inverse transform on the graph, the graph convolution operation can be implemented as follows.

$$x_G^*g = U((U^Tx) \odot (U^Ty)) \quad (9)$$

where $_G^*$ as denotes the graph convolution operator, x denotes the signal in the node domain on the graph, g is the graph convolution kernel, and \odot refers to the Hadamard product, which denotes the multiplication of the corresponding elements of two vectors. By replacing the vector U^Ty with the diagonal array g_θ theta, the Hadamard product is transformed into a matrix multiplication. The graph convolution operation is denoted as $Ug_\theta U^Tx$.

To solve the excessive computation of Laplace eigenvalues and eigenvectors, Defferrar et al. [32] proposed ChebNet based on Chebyshev polynomials. The eigenvalue matrix is approximated by Chebyshev polynomials, and the Chebyshev polynomials are as follows.

$$g_\theta(\Lambda) = \sum_{k=0}^{k-1} \theta_k T_k(\tilde{\Lambda}) \quad (10)$$

where θ is the Chebyshev coefficient; $T_k(\tilde{\Lambda})$ is the k -th order Chebyshev polynomial of $\tilde{\Lambda}$; $\tilde{\Lambda} = \frac{2\Lambda}{\lambda_{max}} - I_n$, $\tilde{\Lambda}$ are the normalized eigenvalue diagonal matrices. Thus, the convolution operation can be expressed as follows:

$$x_G^*g_\theta = U\left(\sum_{k=0}^{k-1} \theta_k T_k(\tilde{\Lambda})\right)U^Tx = \sum_{k=0}^{k-1} \theta_k T_k(\tilde{L})x \quad (11)$$

where $\frac{2L}{\lambda_{max}} - I_n$; the computational complexity of the graph convolution calculation is reduced from $O(N^2)$ to $O(LE)$ by replacing the Chebyshev expansion $T_k(\tilde{\Lambda})$ with the eigen-decomposition part of the frequency domain convolution g_θ in the original GCN, effectively avoiding the computational part of the eigen-decomposition, where E is the number of edges in the input graph and L is the order of the Laplace operator polynomial. ChebNet results in a significant reduction in computational complexity and a significant improvement in computational efficiency.

After that, Kipf et al. used first-order Chebyshev polynomials and simplified the spectral graph convolution by restricting the parameters in order to make ChebNet have better local connectivity properties. Let $K = 2$, $T_0(\tilde{L}) = 1$, $T_1(\tilde{L}) = L$, $\lambda_{max} = 2$. Then the graph convolution calculation is simplified as:

$$x_G^*g_\theta \approx \theta_0 x + \theta_1 (L - I_n)x = \theta_0 x - \theta_1 D^{-\frac{1}{2}} A D^{-\frac{1}{2}} x \quad (12)$$

Where: θ_0 and θ_1 are free parameters, shared by the whole graph. Let $\theta = \theta_0 = -\theta_1$, i.e., the two parameters are transformed into a one-parameter model, then the graph convolution is calculated as:

$$x_{G\theta}^* \approx \theta \left(I_n + D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \right) x \quad (13)$$

However, since $I_n + D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$ is the eigenvalue of $[0, 2]$, which may lead to the problem of disappearing, exploding or unstable values of neural network gradients, $\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}$ is used instead of $I_n + D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$ for normalization.

2.1.2.2 Graph Convolutional Network (DGC) based on spatial domain

The spatial domain approach: spatial-based graph convolutional networks were first proposed in Neural Network for Graphs (NN4G), which is different from the spectral domain graph convolutional neural network from signal processing theory, the spatial domain graph convolutional neural network starts from the nodes in the graph, designs the aggregation function to gather the features of neighboring nodes, adopts the message propagation mechanism, and thinks about how to accurately and efficiently use the features of neighboring nodes of the central node to update the features of the central node. The essence of CNN is weighted summation, and the spatial domain graph convolutional neural network is based on the basic construction process of CNN to accomplish the purpose of GNN aggregation of neighboring nodes from the perspective of summation. Since the nodes in the graph are unordered and the number of neighboring nodes is uncertain, one idea of the spatial domain graph convolutional neural network is (1) to fix the number of neighboring nodes and (2) to sort the neighboring nodes. If the above two tasks are completed, the non-Euclidean structured data becomes ordinary Euclidean structured data, and naturally the traditional algorithm can be completely migrated to the graph. Among them, step (1) also facilitates the application of GNN to graphs with many nodes.

Currently, GCN has become a fundamental model for traffic flow prediction research and a benchmark method for experiments. Although neither the air-domain graph convolution network nor the frequency-domain graph convolution network is proposed for the traffic flow prediction problem, the natural graph structure property of traffic data makes GCN show high efficiency and accuracy in the field of traffic flow prediction than the traditional methods.

2.2 Modeling experiments for traffic prediction

This section will first give a specific definition of the traffic flow prediction problem and then give the flow of the traffic flow prediction model based on spatiotemporal characteristics.

2.2.1 Construction of traffic road network graph structure

Traffic prediction is a typical time series prediction problem [33], and its road network traffic flow data exhibits a high degree of periodicity, which provides a great deal of potential for traffic prediction. **Figure 6** shows the traffic data for the first week of December for individual toll stations on the Shaanxi Provincial Free-way, demonstrating a high degree of periodicity.

Given the first M flow observations, the flow data measured at the n sensor stations at time step H can be viewed as a matrix of size $M \times N$. The most likely flow measurements predicted at the next H time steps are:

$$F_{t+1}, \dots, F_{t+H} = \operatorname{argmax} \log P(F_{t+1}, \dots, F_{t+H} | F_{t-M+1}, \dots, F_t) \quad (14)$$

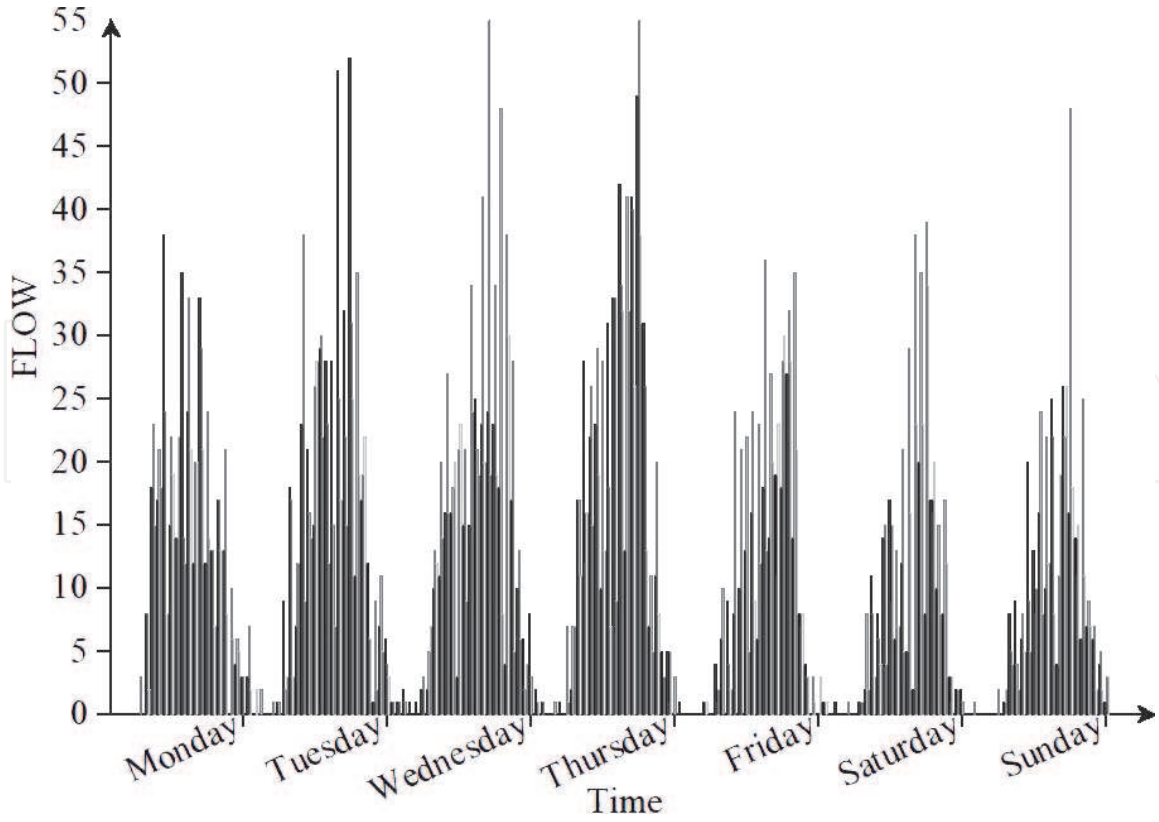


Figure 6.
Periodicity of traffic data.

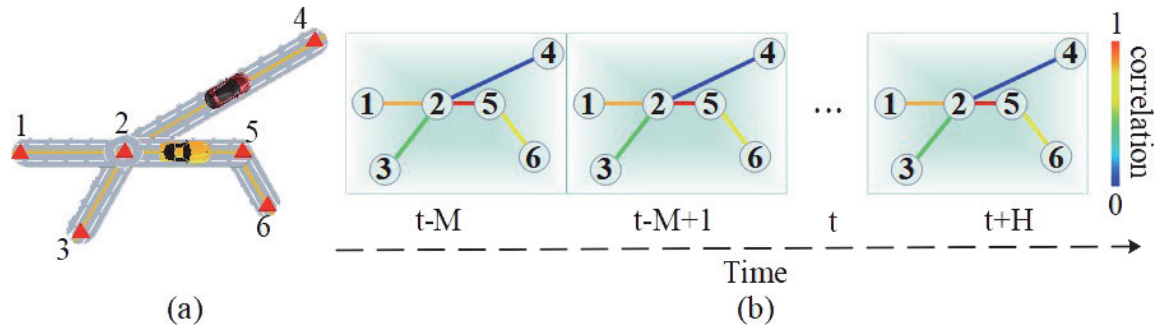


Figure 7.
Spatiotemporal correlations. (a) Stations in a road network. (b) Dynamic spatial correlations.

where $F_t \in R^n$ is a vector of observations for n road segments at time step t , where each element records the historical observations for a single road segment. For unordered road network traffic data, the observations F_t are not independent and can be viewed as graph signals defined on an undirected graph G with weight as shown in **Figure 7**, the graph is expressed in terms of an adjacency matrix $G_t = (V_t, E, W)$. F_t is a finite set of vertices corresponding to the observations of the n toll stations in the traffic. E is a set of edges representing the connections between stations, and $W \in R^{n \times n}$ represents the weighted adjacency matrix of G_t .

2.2.2 Traffic data acquisition and preprocessing

2.2.2.1 Traffic datasets

Traffic flow prediction by deep learning requires a large amount of data support, that is, real-world road traffic speed data. With the continuous improvement of

traffic facilities, the amount of traffic data has also produced an explosive growth. Traffic flow prediction is precisely based on huge traffic data, so understanding the current common traffic data is the basis for achieving traffic flow prediction. The sources of traffic data mainly include road fixed-point detectors, vehicle GPS records, bus IC cards, license plate recognition, cell phone data, etc. We have made the common traffic data used for traffic flow prediction as **Table 1**.

2.2.2.2 Data preprocessing

Traffic flow data mainly detects parameters such as speed, flow rate, time, etc. The data collection process may result in detection equipment failure, instrument error, software failure, communication interference, environmental noise, etc., and even sudden road failure may have a great impact on the data, resulting in real-time data may be missing or abnormal, so the overall process of validity processing of this type of traffic data according to its type is shown in **Figure 8**.

- Abnormal data processing

The preprocessing methods of abnormal data can be divided into two categories: Data rejection. Data rejection can be used when there is less erroneous data in the traffic data. The rejection of individual erroneous data will not affect the integrity and trend of the data, but if the proportion of erroneous data is large, the rejection method cannot be adopted because too much rejection of erroneous data will

Data set classification	Data set name	Data fields	Sampling period
Expressway	PeMS	Timestamp, Station ID, Region. Highway ID, direction, trip	5 min
	METR-LA	Vehicle speed	5 min
	SEATTLE LOOP	Vehicle speed	5 min
	Madird Traces	Vehicle track	0.5 s
	Los-Loop	Vehicle speed	5 min
Cabs	NYC Taxi	Boarding and alighting times, location, distance traveled, fare, payment type	/
	TaxiBJ21	GPS data and weather data	30 min
	SH-Speed	Vehicle ID, location, operation status, speed	10 min
	CRAWDAD	Vehicle ID, time, coordinates	7 s
	SZ-taxi	Vehicle speed	15 min
	T-Drive	Vehicle track	/
Internet taxi	Didi-GAIA-Open-Data	Vehicle speed	/
Rail Transit	SHMetro HZMetro	People flow	15 min
City Road Network	VTC (vant-trace-creteil)	Time, lane, vehicle angle, speed, and vehicle ID	1 s
	Od_bologna Koln.Tr	Coordinates, speed, vehicle ID	1 s
	NYC-Bike	Vehicle ID, coordinates, time	/

Table 1.
Common data sets for traffic flow prediction models.

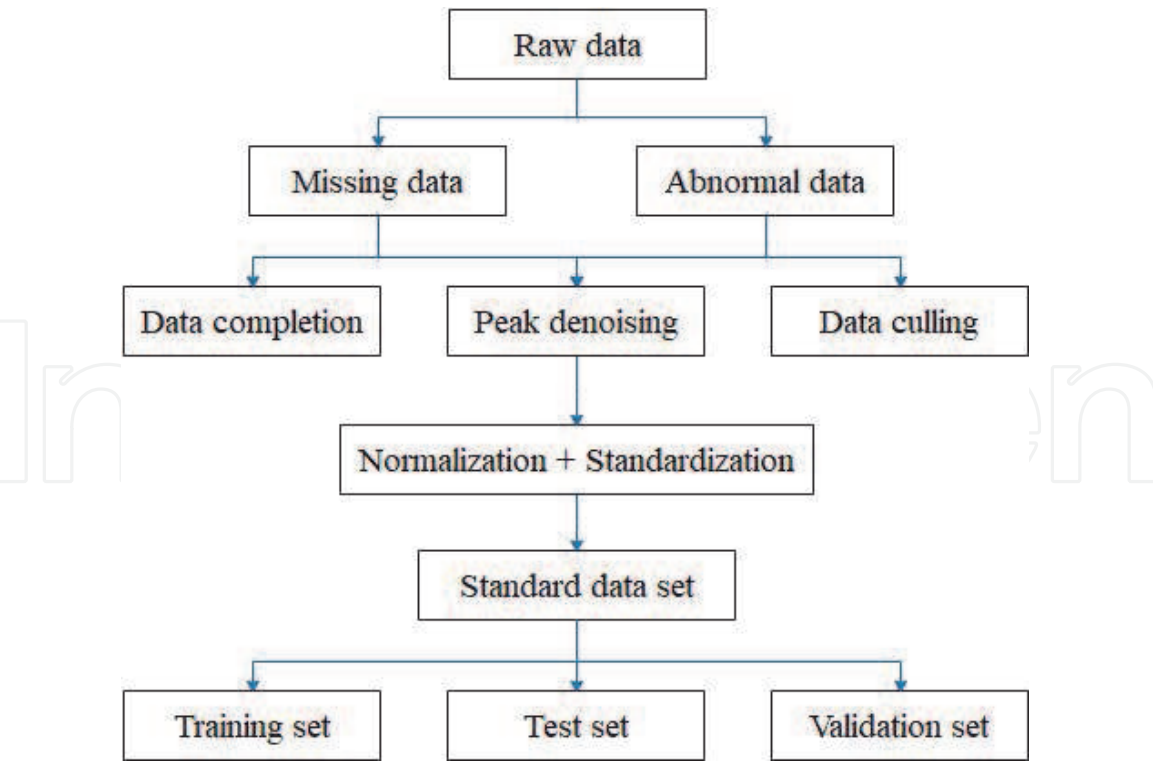


Figure 8.
Overall flow of data preprocessing.

destroy the integrity of the data and its trend. Peak denoising. Since traffic data is highly nonlinear and the traffic data at peak hours can be very significant, i.e., the noise oscillation region during peak hours, peak denoising is needed. Commonly used methods such as empirical mode decomposition (EMD), i.e., fluctuation decomposition in the local oscillation part of the trend change.

- Missing data processing [34]

Missing data is caused by hardware and software factors that do not detect data at the detection end or packet loss during data communication. In road traffic, this can be due to excessive vehicle density and inaccurate data collection by traffic flow detection instruments, data failures in transmission, and many other reasons for gaps in the collected data, such as missing data at a point in time, a certain period, or several periods of time. Typically, there are two classical missing patterns in time series data as shown in **Figure 9** below. **Figure 9(a)** indicates that the exported toll records have randomly lost observations at a single toll station, and the white circles indicate the missing values. **Figure 9(b)** indicates that there are several consecutive time points in the records of multiple toll stations with no observed values, which is

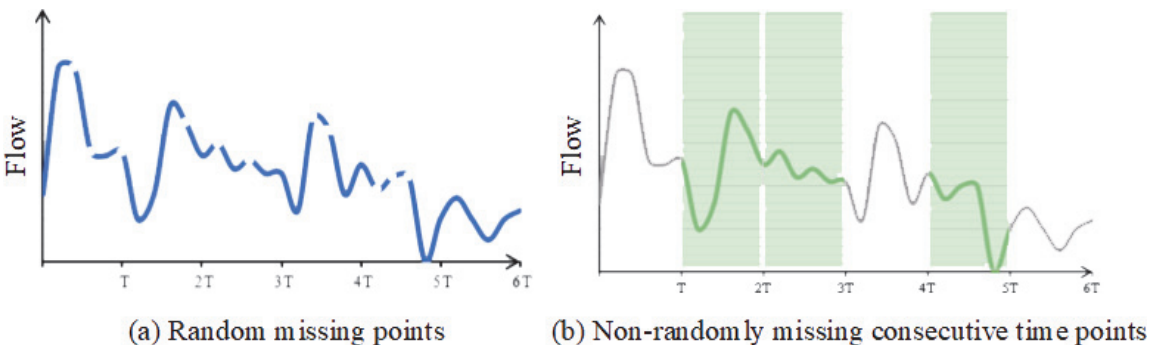


Figure 9.
Example of missing pattern of spatiotemporal data (traffic data as an example).

a more common pattern of missing spatiotemporal traffic data. The green curve in the green panel represents the observed values and the gray curve represents the missing values. This situation requires correlating and processing the missing data, and then repairing the data using interpolation and smoothing algorithms, prior to dimensionlessizing the data using initialization operators to consider the fact that the units and orders of magnitude of the characteristic series of influencing factors are not uniform.

- Data normalization/normalization

Generally, the obtained traffic data are scattered, and the distribution characteristic curve presented by the data is fuzzy, and the distribution cannot be determined. Therefore, the data do not satisfy the normal distribution and need to be normalized to regularize the data and improve the comparability between the data to facilitate the subsequent model prediction. The data are z-core normalized to approximately satisfy the normal distribution, so that the weights are more evenly distributed in the subsequent model training, i.e.,

$$x' = \frac{x - \mu}{\sigma} \tag{15}$$

where μ, σ are the mean and standard deviation.

2.2.3 Classical graph convolution framework

To solve the problem of non-Euclidean structure of traffic network data, graph neural networks are often used to model spatial dependencies in traffic networks, and then convolution is used to fundamentally improve the efficiency of graph analysis and network construction from frequency and spatial domains, i.e., Graph Convolutional Network (GCN). Graph Convolution extends traditional convolution to graph-structured data, and powerful methods such as graph convolutional networks and their variants are widely used for these spatiotemporal network data prediction tasks with good performance. Most existing graph convolutional traffic flow forecasts are spatiotemporal in nature, since most traffic data sets have both spatial and temporal attributes. The development of traffic flow prediction models based on graph convolutional networks is presented as in **Figure 10**. In this paper, five of the most typical and most referenced models will be selected for illustration.

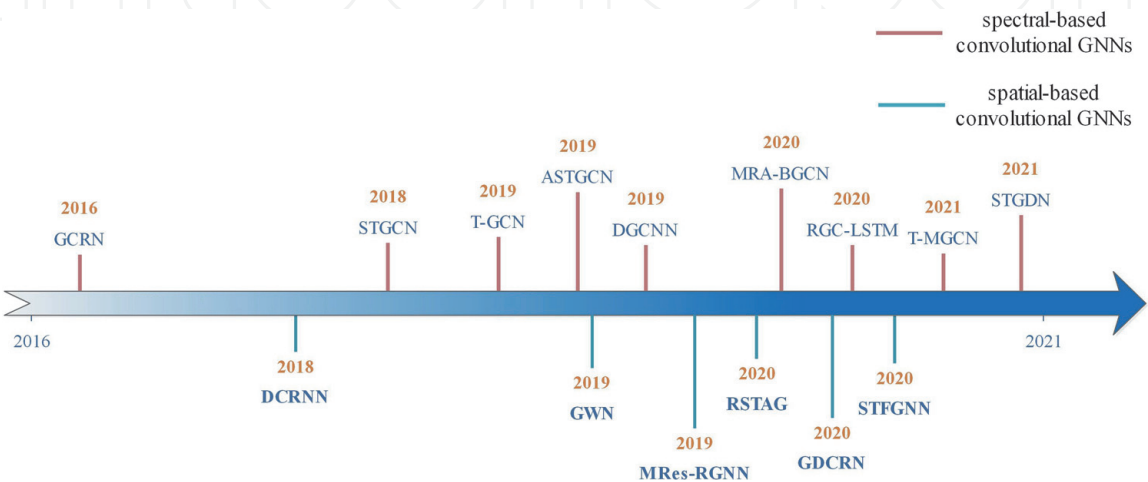


Figure 10.
Traffic flow prediction model based on graph convolution.

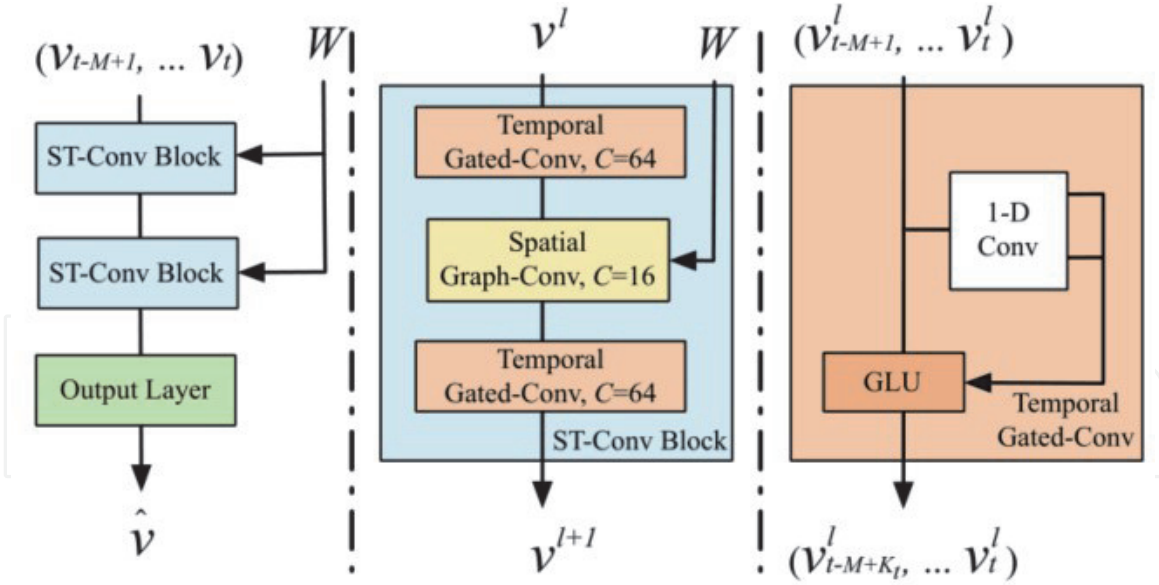


Figure 11.
Architecture of spatiotemporal graph convolutional networks.

2.2.3.1 STGCN predicts traffic flow

The STGCN model proposed by Yu et al. [19] (**Figure 11**) (2018AAAI) for the first time uses graph structures to model traffic networks while using graph convolution to model spatiotemporal sequences and uses pure convolutional structures to extract spatiotemporal features from the graph structures simultaneously.

STGCN is composed of several spatiotemporal convolutional blocks, each of which is formed as a “sandwich” structure with two gated sequential convolution layers and one spatial graph convolution layer in between. The framework STGCN consists of two spatiotemporal convolutional blocks (ST-Conv blocks) and a fully-connected output layer in the end. Each ST-Conv block contains two temporal gated convolution layers and one spatial graph convolution layer in the middle. The residual connection and bottleneck strategy are applied inside each block.

The model, although using convolution instead of LSTM-like patterns, does speed up training, but it also leads to missing historical data information and can only achieve short-term prediction, not long-term prediction, and graph convolution captures information between different nodes to model spatial models, which does not seem to make good use of the potential relationships between different regions.

2.2.3.2 DCRNN for predicting traffic flow

The DCRNN model proposed by Li et al. [22] (**Figure 12**) (2018ICLR) models spatial correlation as a diffusion process on directed graphs, thus modeling the transformation of traffic flow, and proposes diffusion convolution recurrent neural networks that can capture the spatial and temporal dependence between time series using a framework of seq2seq. To address these challenges, we propose to model the traffic flow as a diffusion process on a directed graph and introduce Diffusion Convolutional Recurrent Neural Network (DCRNN), a deep learning framework for traffic prediction that incorporates both spatial and temporal dependency in the traffic flow. Specifically, DCRNN captures the spatial dependency using bidirectional random walks on the graph and the temporal dependency using the encoder-decoder architecture with scheduled sampling.

2.2.3.3 **GMAN** prediction of traffic flow

The GMAN model (2020AAAI) proposed by Zheng et al. [23] (**Figure 13**) uses a spatiotemporal attention mechanism to model dynamic spatial relationships and nonlinear temporal relationships separately, while using a gating mechanism to

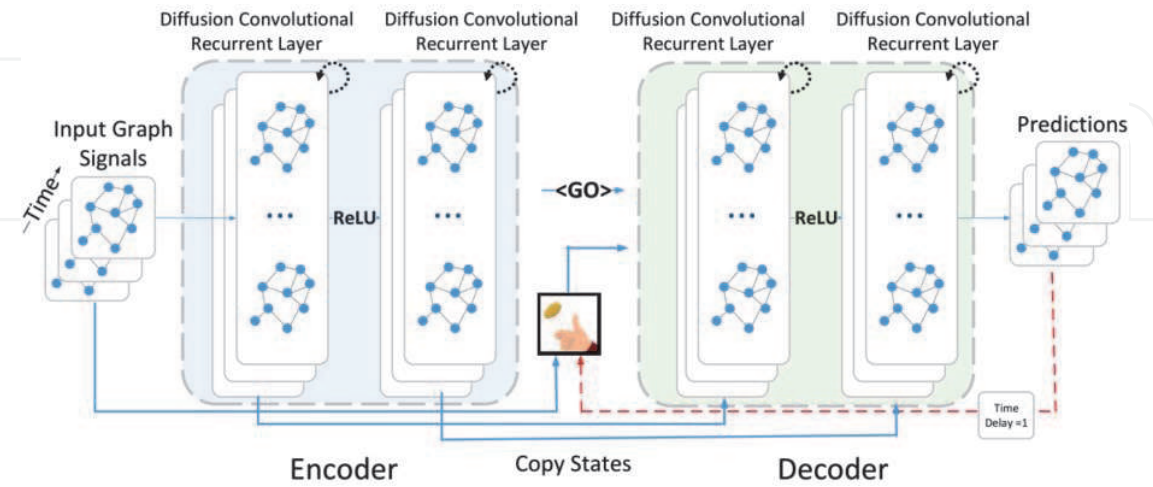


Figure 12. System architecture for the diffusion convolutional recurrent neural network designed for spatiotemporal traffic prediction.

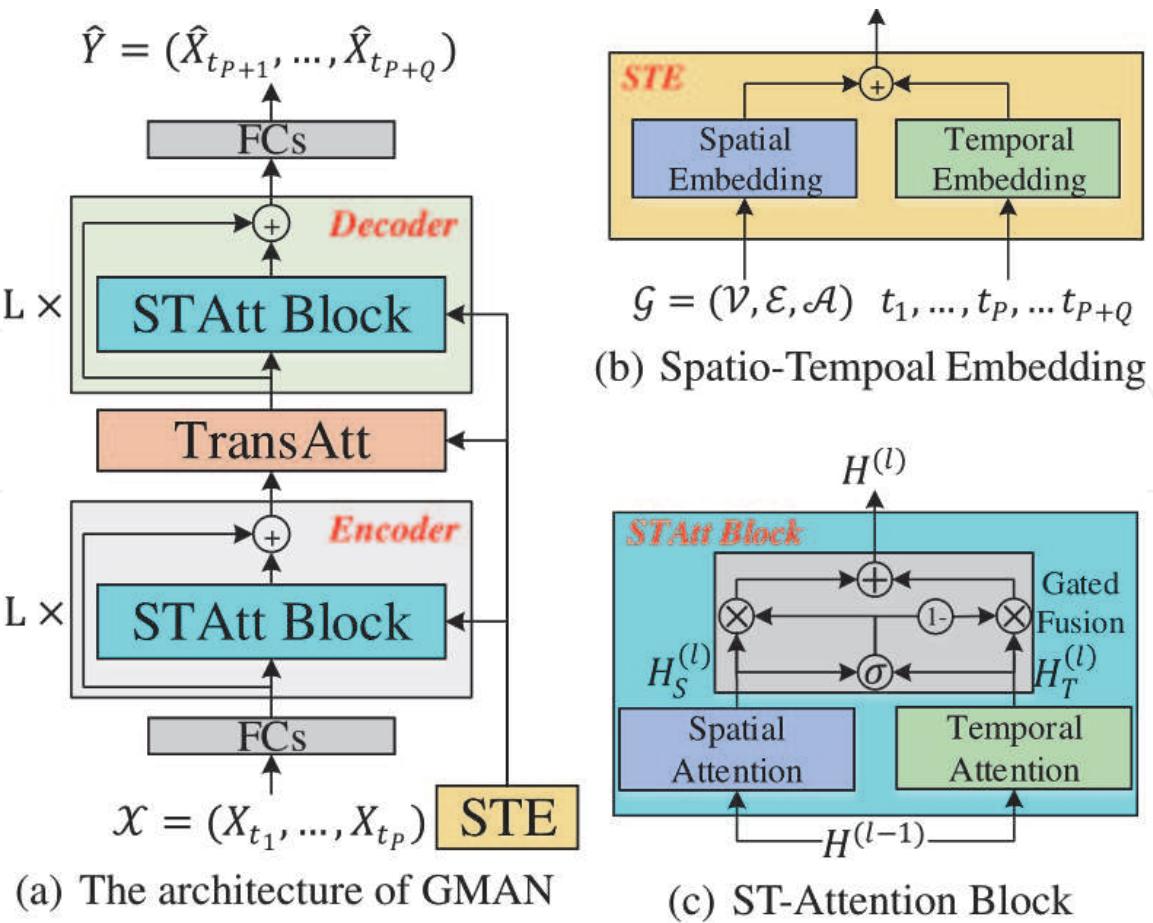


Figure 13. The overall framework of GMAN model. (a) the framework of GMAN. (b) Spatiotemporal embedding. (c) the ST-attention block.

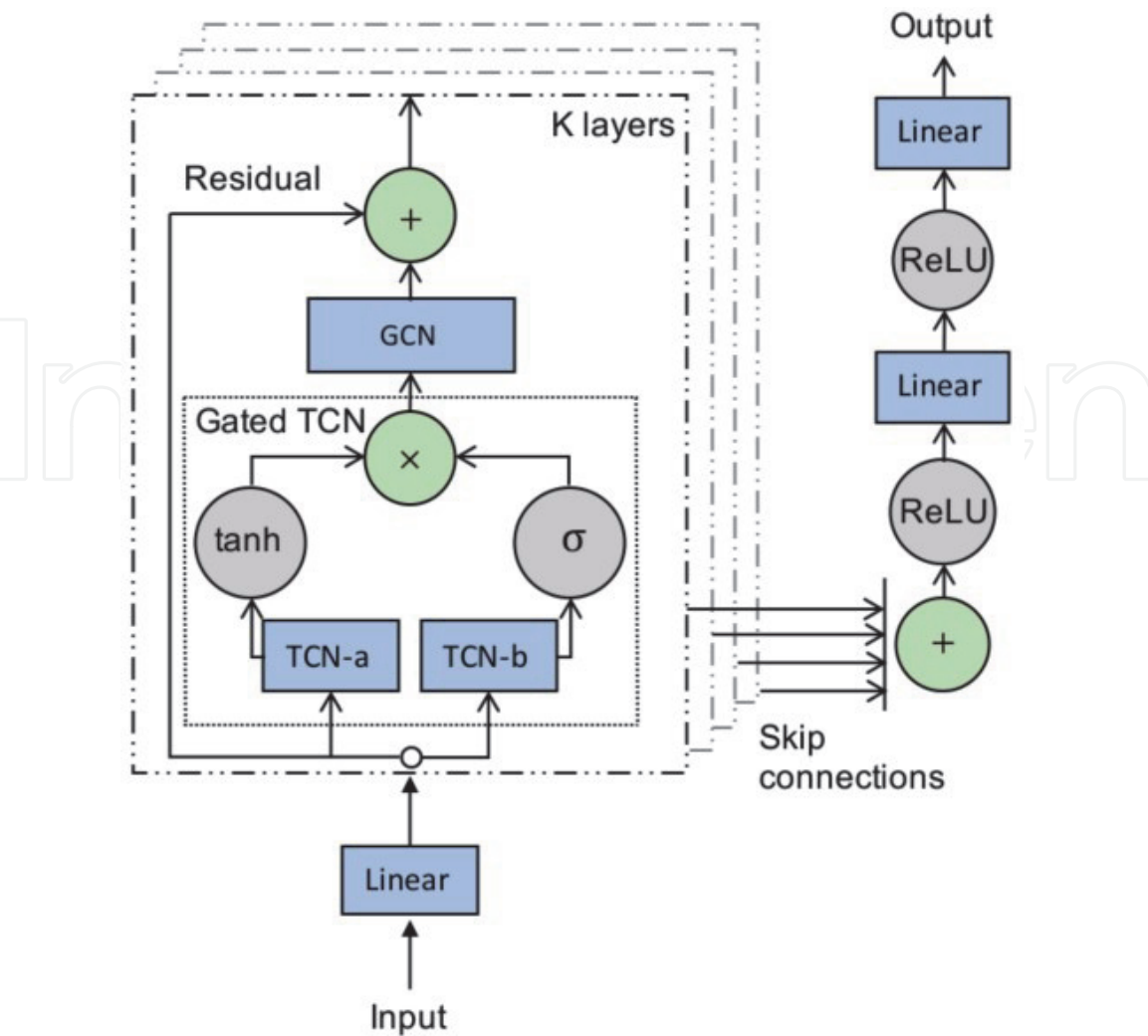


Figure 14.
The framework of graph WaveNet.

adaptively fuse the information extracted by the spatiotemporal attention mechanism.

Because the whole traffic is a network, the error of one node is amplified by other nodes, which affects the final prediction results. To solve the above problem, GMAN adopts an encoder-decoder architecture, where encoder is used to extract features and decoder to predict. A transformed attention layer is applied in between these two to transform the encoded traffic features to generate a sequential representation of future time steps as the input to the decoder. Here Encoder and Decoder are composed of ST-attention block. Then the authors use an STE block to combine the spatial and temporal information and then input into the ST-ATTENTION block to solve the problem of complex time-space correlation. Finally, the experimental results of the article on two real-world traffic prediction tasks (i.e., traffic volume prediction and traffic speed prediction) demonstrate the superiority of GMAN.

2.2.3.4 Graph WaveNet prediction of traffic flow

Wu et al. [35] (**Figure 14**) propose in this paper a novel graph neural network architecture, Graph WaveNet, for spatial-temporal graph modeling. The model uses the idea of diffusion convolution in extracting spatial features of road networks and adds a novel adaptive connection matrix to make up for the deficiency of fixed

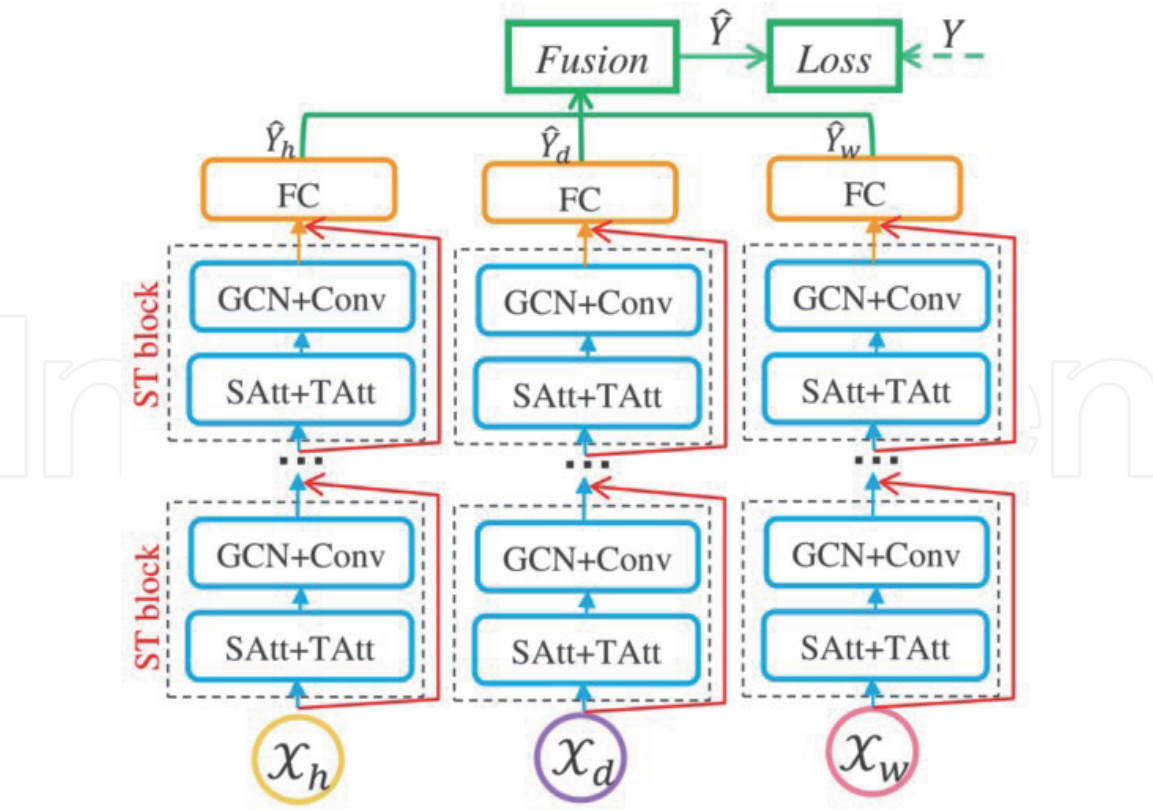


Figure 15.
The framework of ASTGCN. SAtt: Spatial attention.

topology in extracting spatial features and employs dilated causal convolution and gate mechanism on time series without the traditional RNNs cycle, which is validated by METR-LA and PEMS-BAY data sets, GWN in terms of training effect and time good results were achieved.

2.2.3.5 ASTGCN prediction of traffic flow

Guo et al. [24] (**Figure 15**) propose a novel attention-based spatial-temporal graph convolutional network (ASTGCN) model to solve traffic flow prediction problem. ASTGCN mainly consists of three independent components to respectively model three temporal properties of traffic flows, i.e., recent, daily-periodic, and weekly-periodic dependencies. More specifically, each component contains two major parts: 1) the spatial-temporal attention mechanism to effectively capture the dynamic spatial-temporal correlations in traffic data; 2) the spatial-temporal convolution, which simultaneously employs graph convolutions to capture the spatial patterns and common standard convolutions to describe the temporal features. The output of the three components is weighted fused to generate the final prediction results.

These five typical traffic flow prediction models above are compared in **Table 2**.

2.3 Application scenarios for traffic flow prediction

Many researchers have already applied the proposed traffic flow prediction models to various traffic scenarios and achieved excellent results. For example, by predicting the traffic flow of a roadway in advance, it can provide drivers with more advanced travel routes. In addition, it can also provide prerequisites for traffic light optimization, etc. In this paper, three scenarios are chosen to illustrate the

Models	Model's characteristics
STGCN (2018)	(1) Compared with traditional spatio-temporal models (RNN, LSTM) based on recurrent neural networks, the STGCN model combines graph convolution and gated time convolution, and for the first time uses pure convolutional layers to extract time and space information at the same time. (2) The STGCN model uses one-dimensional convolution to learn information in the time dimension and is not limited by the prediction data at the previous time point, so that the model can better capture the drastic changes in the data (such as traffic flow data during peak hours). (3) Due to the characteristics of the convolutional structure, STGCN model is parallelized at the input, with fewer parameters and faster training speed, allowing the model to process large-scale networks with higher efficiency.
DCRNN (2018)	(1) In view of the dynamic characteristics of traffic flow, DCRNN model introduces diffusion convolution when modeling spatial dependence and considers forward propagation and back propagation and is more suitable for traffic networks with a directed graph structure. (2) When the DCRNN model models the time dependence, the matrix multiplication in the GRU model is changed to diffusion convolution, and the diffusion convolution-gated recurrent unit (DCGRU) is obtained.
ASTGCN (2019)	(1) The ASTGCN model effectively learns the dynamic spatiotemporal correlation of traffic data through the spatiotemporal attention mechanism. (2) The ASTGCN model designed a multichannel network structure from multiple time periods, combining global time and space information to improve prediction accuracy
GWN (2019)	(1) In terms of spatial dependency acquisition, The GWN model constructs an adaptive dependency matrix that can retain the implicit spatial relationship of the road network. (2) In the acquisition of time dependence, The GWN model adopts stacked dilated 1D convolution and does not need to consider the problem of information disappearance too long ago and can extract longer time dependence than RNN-based cyclic convolutional networks.
GMAN (2020)	(1) The GMAN model refines the complexity of time and space. It is divided into dynamic spatial correlations and Non-linear temporal correlations, and the attention mechanism is introduced. (2) The GMAN model solves the cumulative error problem of stepwise prediction. A transformation attention layer is added between the encoder and the decoder, so that historical and predicted traffic characteristics can be converted.

Table 2.
Characteristics of typical models.

application of traffic flow prediction in the context of highways. These three scenarios are the work that has been done by our team so far, and the reason for choosing the highway is that the work in this area is more mature.

2.3.1 Scenario I. quantitative assessment on truck-related road risk for the safety control

Traffic conditions of truck flow is one of the critical factors influencing transportation safety and efficiency, which is directly related to traffic accidents, maintenance scheduling, traffic flow interruption, risk control, and management. The estimation of the truck flow of various types could be better to identify the irregular flow variation introduced by various trucks and quantitatively assessed the corresponding road risks.

Jin et al. [36] first improved on the gated recursive unit (GRU) based on a deep learning approach to estimate various types of truck traffic. Then a multiple logistic regression method was proposed to classify the road risk into three classes: safe, risky, and dangerous. According to the CSV trend, road risks are classified into

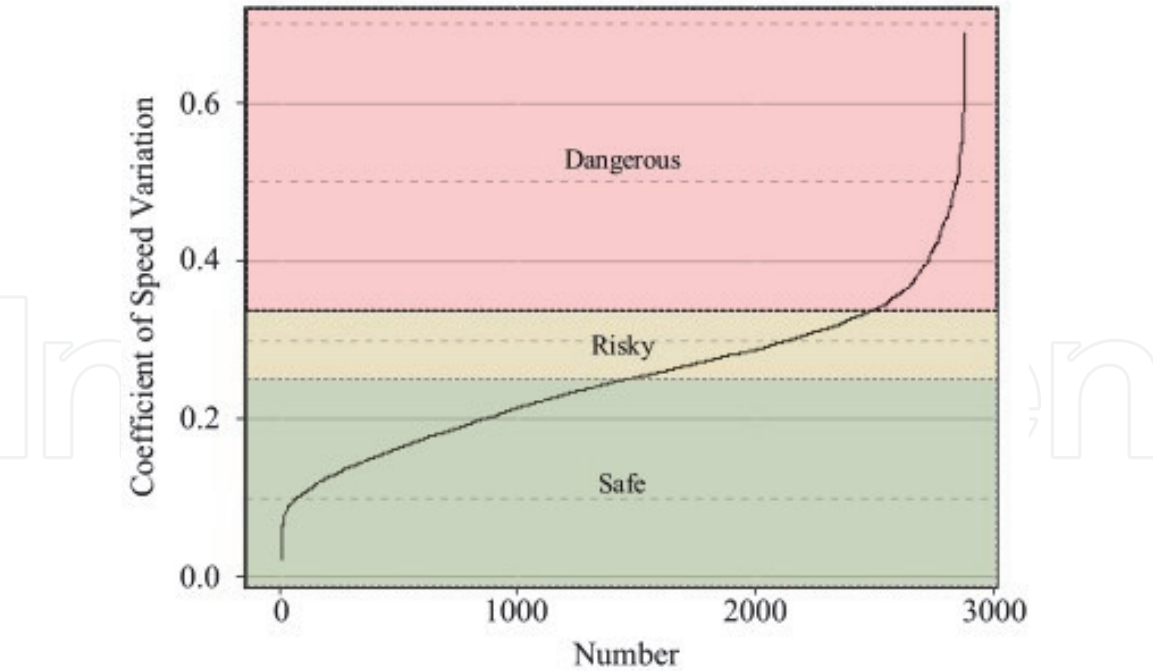


Figure 16.
Coefficient of speed variation (CSV) of passenger cars.

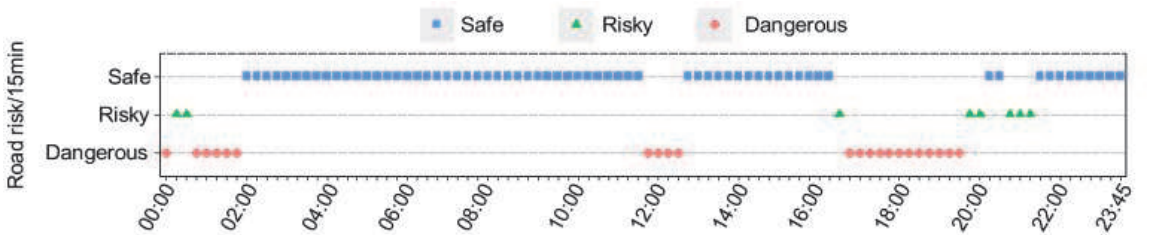


Figure 17.
Road risk assessed by predicted truck flow in April 12, 018.

three categories as shown in **Figure 16**. Different risk classes can guide traffic control and management and broadcast traffic information to drivers to help them choose their travel routes.

Finally, the road risk calculated by the predicted truck traffic is shown in **Figure 17**, from which the road risk status can be obtained at every moment.

2.3.2 Scenario II. Improved manpower planning for highway toll gate

In China, the relatively heavy queues at freeway toll booths and service areas during peak hours, coupled with the saturation of manpower scheduled during off-peak hours, are undoubtedly a huge obstacle to efficient and cost-effective freeway operations. Therefore, it needs an intelligent manpower planning strategy to simultaneously ensure the efficiency of highway transportation management and road user satisfaction.

Jin et al. [37] addressed a high-precision prediction of vehicle flow based on historical multisource traffic data. Based on the prediction results, an improved manpower planning strategy is proposed to schedule the work accordingly. And the method was tested on a randomly selected toll station as an example, as **Figure 18** shows the daily traffic pattern of the highway Hechizhai toll station.

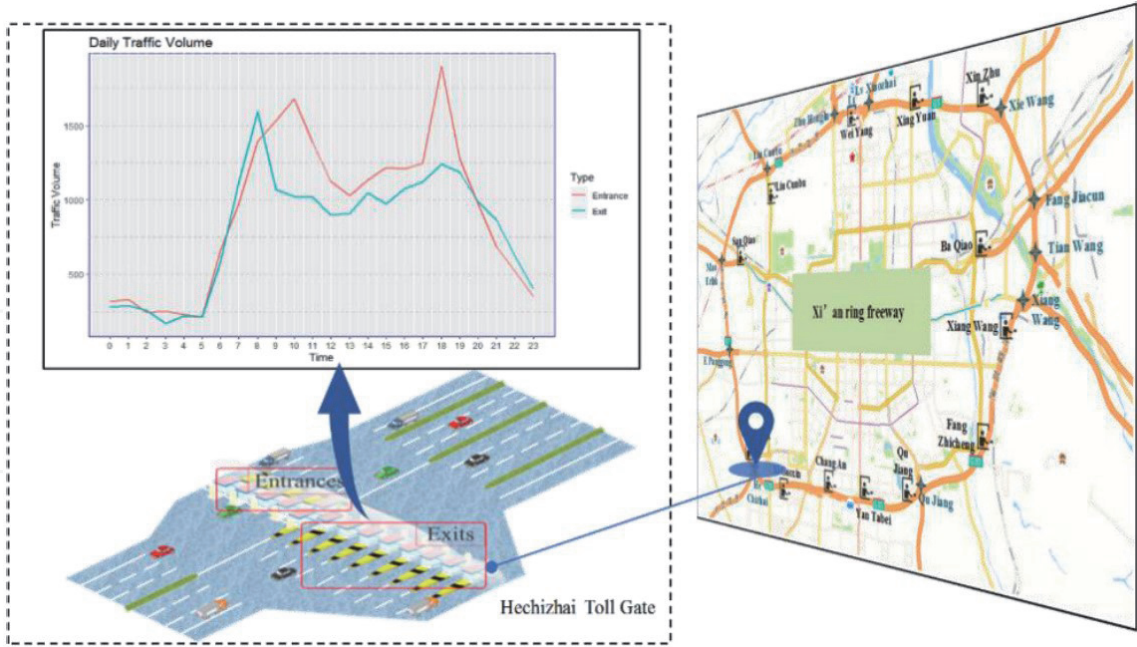


Figure 18.
Daily traffic pattern of the Hechizhai toll gate on the freeway.

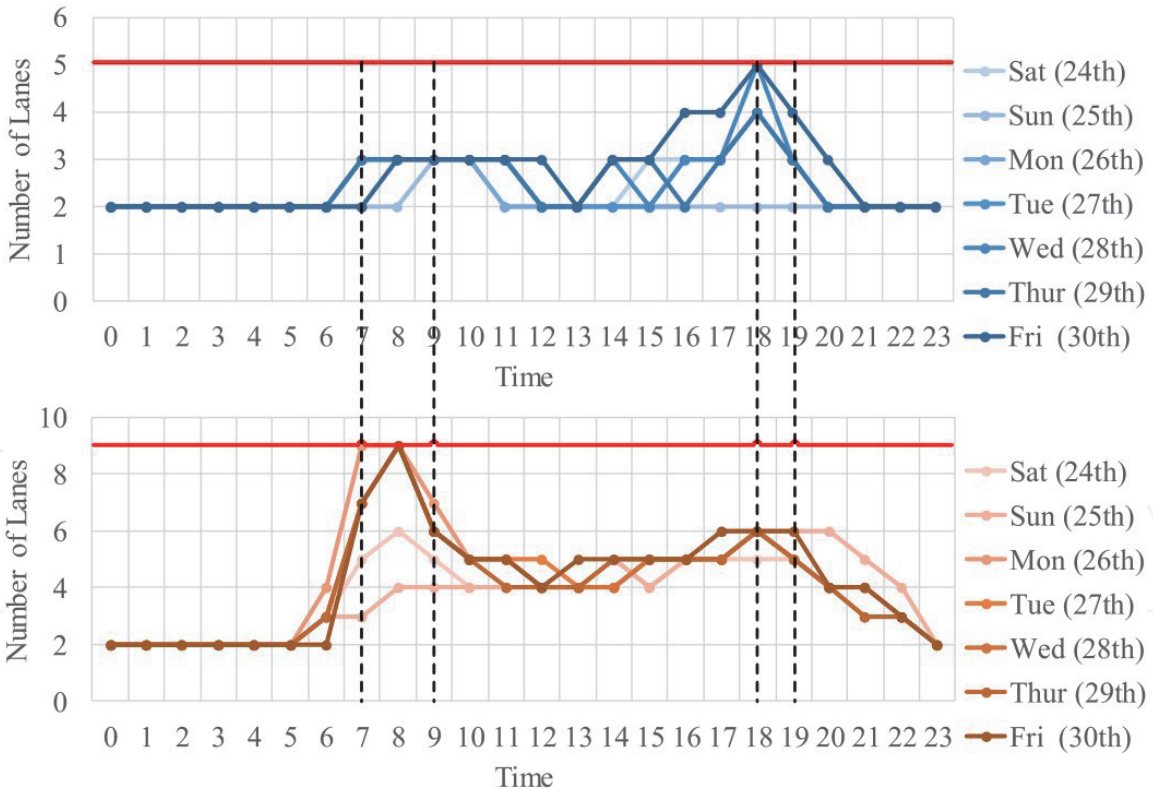


Figure 19.
Reversible toll lane configuration suggestion. The red lines refer to the number of toll lanes designed in each direction. The blue and brown gradient lines indicate the change in the number of exit and entrance lanes per hour, respectively. The black dash lines depict the reversible toll lane change period.

The results show from **Figure 19** that the upper part and the lower part show the lane opening at the entrance and exit of toll gates in one week, respectively. Two narrow black dotted lines indicate the morning and evening peak hours. During the morning peak, it is obvious that two entry lanes are not used while the number of toll lanes of the exit has reached its upper threshold. The opposite phenomenon is



Figure 20.
Comparison of duplex lane simulation process.

seen in the evening peak. Therefore, we suggest that one or two of the entrance lanes can be set as a reversible lane so that the traffic pressure can be released in peak hours. Note that the usage condition of this suggestion is that the entrance and exit of the toll gate must be adjacent such as Hechizhai toll gate.

2.3.3 Scenario III. Capacity analysis of toll stations

Yuan et al. [38] used the results of traffic prediction to analyze the capacity of a toll station and used different queuing models to describe the capacity of typical lanes and compared the delay time and queue length of each model and obtained that the single-way model is more efficient in a typical system. The traffic index of the multiplex lane is also simulated, and the specific simulation process is shown in **Figure 20**, and the capacity of the multiplex lane is obtained to be larger than that of the typical MTC lane, which can relieve the traffic pressure during the peak hours.

3. Conclusions

Traffic is the main driving force of urban development, and real-time and accurate traffic flow prediction is the key to the application of intelligent transportation system. Graph convolutional neural network is an efficient model for processing graph data and has received a lot of attention from researchers in the past few years. This section attempts to summarize the recent graph convolutional neural network models and their applications to traffic flow prediction.

1. This section summarizes the GCN-based traffic flow prediction model. Starting from the basic definition of graph convolution, the basic principles of GCN are introduced with the focus on frequency-domain graph convolution

and space-domain graph convolution. Then, the representative models are clarified, and the structure and characteristics of different prediction models are further categorized and reviewed.

2. This section provides the traffic prediction problem with constructed traffic graph structure. Some public traffic data sets that are widely used in scientific research worldwide are introduced for traffic prediction experiments, including their data sources, data contents, and data acquisition addresses, and the whole data processing process is analyzed.
3. The application scenarios of traffic flow prediction are discussed. There are two applications are provided: 1) with prediction of the traffic flow of truck, the transportation safety and efficiency could further assess. 2) The work schedule arrangement could also improve based on the prediction of traffic flow to avoid manpower waste and allow more passing gate to open in the peak hours. Other than those applications, there still many aspects worth to explore.

Acknowledgements

Thanks to the following researchers for their great support in the writing of this book, especially Wenbang Hao, Erlong Tan, Wanrong Xu, Zhen Jia, Yiwen Gao, Yajie Zhang, etc., who have put a lot of energy into many formulas and illustrations.


This work was supported in part by the National Key R&D Program of China (2020YFB1600400), Key Research and Development Program of Shaanxi Province (No.2020GY-020), National Natural Science Foundation of China (Grant No. 51505037) and Supported by the Fundamental Research Funds for the Central Universities, CHD (Grant No. 300102320305).

Author details

Ping Wang*, Tongtong Shi, Rui He and Wubei Yuan
Chang'an University, Xian, China

*Address all correspondence to: pingwang@chd.edu.cn

IntechOpen

© 2022 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Wei Chen H, Klaiber A. Does road expansion induce traffic? An evaluation of vehicle-kilometers traveled in China. *Journal of Environmental Economics and Management*. 2020;**104**:95-696. DOI: 10.1016/j.jeem.2020.102387
- [2] Ye J, Zhao J, Ye K, Chengzhong X. How to build a graph-based deep learning architecture in traffic domain: A survey. *IEEE Transactions on Intelligent Transportation Systems*. 2020;**2**(6):1-20. DOI: 10.1109/TITS.2020.3043250
- [3] Wang P, Hao W, Jin Y. Fine-grained traffic flow prediction of various vehicle types via fusion of multisource data and deep learning approaches. *IEEE Transactions on Intelligent Transportation Systems*. 2020;**5**: 8-10. DOI: 10.1109/TITS.2020.2997412
- [4] Tian Z, Jia L, Dong H, Zhang Z. Determination of key nodes in urban road traffic network. Shenyang, China: Proceeding of the 11th World Congress on Intelligent Control and Automation. 29 June-4 July 2014, IEEE; 2014. pp. 3396-3400. DOI: 10.1109/WCICA.2014.7053279
- [5] Wang P, Xu W, Jin Y, Wang J, Li L. Forecasting traffic volume at a designated cross-section location on a freeway from large-regional toll collection data. *IEEE Access*. 2019;**7**: 9057-9070. DOI: 10.1109/ACCESS.2018.2890725
- [6] Jin Y, Xu W, Wang P. SAE Network: A Deep Learning Method for Traffic Flow Prediction. 2018 5th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS). 2018:241-246. DOI: 10.1109/ICCSS.2018.8572451
- [7] Yu X, Sun L, Yang Y, Liu G. A short-term traffic flow prediction method based on spatial-temporal correlation using edge computing. *Computers & Electrical Engineering*. 2021;**93**:107219. DOI: 10.1016/j.compeleceng.2021.107219
- [8] Miska MP. Microscopic Online Simulation for Real-Time Traffic Management. TRAIL Research School. 2007
- [9] Ngoduy D, Wilson RE. Multianticipative nonlocal macroscopic traffic model. *Computer-Aided Civil and Infrastructure Engineering*. 2014; **29**(4):248-263
- [10] Okutani I, Stephanedes YJ. Dynamic prediction of traffic volume through Kalman filtering theory. *Transportation Research Part B: Methodological*. 1984; **18**(1):1-11
- [11] Kumar SV, Vanajakshi L. Short-term traffic flow prediction using seasonal ARIM A model with limited input data. *European Transport Research Review*. 2015;**7**(3):21
- [12] Jeong YS, Byon YJ, Castro-Neto MM, Easa SM. Supervised weighting-online learning algorithm for short-term traffic flow prediction. *IEEE Trans. on Intelligent Transportation Systems*. 2013;**14**(4):1700-1707. DOI: 10.1109/TITS.2013.2267735
- [13] Zhu S, Lin C, Chu Z. Bayesian network model for traffic flow estimation using prior link flows. *Journal of Southeast University (English Edition)*. 2013;**29**(3):322-327. DOI: 10.3969/j.issn.1003-7985.2013.03.017
- [14] Qi Y, Ishak S. A Hidden Markov Model for short term prediction of traffic conditions on freeways. *Transportation Research Part C: Emerging Technologies*. 2014;**43**:95-111. DOI: 10.1016/j.trc.2014.02.007

- [15] Chen M, Yu X, Yang L. PCNN: Deep convolutional networks for short-term traffic congestion prediction. *IEEE Transactions on Intelligent Transportation Systems*. 2018;**19**(11): 3550-3559. DOI: 10.1109/TITS.2018.2835523
- [16] Lv Y, Duan Y, Kang W, Li Z. Traffic flow prediction with big data: A deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*. 2015;**16**(2):865-873. DOI: 10.1109/TITS.2014.2345663
- [17] Hochreiter S, Schmidhuber J. Long short-term memory [J]. *Neural Computation*. 1997;**9**(8):1735-1780. DOI: 10.1162/neco.1997.9.8.1735
- [18] Cho K, van Merriënboer B, Gulcehre C, Bahdanau D. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2014
- [19] Yu B, Yin H, Zhu Z. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*. 2018. pp. 3634-3640
- [20] Zhang J, Yu Z, Qi D. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI'17)*. AAAI Press; 2017. pp. 1655-1661
- [21] Pei H, Wei B, Chang KC-C, Lei Y, Yang B. Geom-GCN: Geometric Graph Convolutional Networks[J]. *arXiv preprint arXiv:2002.05287*, 2020
- [22] Li Y, Yu R, Shahabi C, Liu Y. Diffusion convolutional recurrent neural network: data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*, 2017
- [23] Zheng C, Fan X, Cheng W, Qi J. GMAN: A graph multi-attention network for traffic prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2020;**34**(01): 1234-1241. DOI: 10.1609/aaai.v34i01.5477
- [24] Guo S, Lin Y, Feng N, Song C, Wan H. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2019;**33**:922-929. DOI: 10.1609/aaai.v33i01.3301922
- [25] Kipf T, Welling M. Semi-supervised classification with graph convolutional networks. Published as a conference paper at ICLR. 2017. *arXiv:1609.02907*
- [26] Liu Z, Zhou J. Introduction to Graph Neural Networks. *Synthesis Lectures on Artificial Intelligence and Machine Learning*. 2020;**14**(2):1-127
- [27] Cheng Y, Liu Z, Cunchao T, Shi C, Sun M. Network Embedding: Theories, Methods, and Applications[J]. *Synthesis Lectures on Artificial Intelligence and Machine Learning*. 2021;**15**(2):1-242. DOI: 10.2200/S01063ED1V01Y202012AIM048
- [28] Vassilis NI, Marques AG, Giannakis GB. Tensor Graph Convolutional Networks for Multi-Relational and Robust Learning. *IEEE Transactions on Signal Processing*. 2020;**68**:6535-6546. DOI: 10.1109/TSP.2020.3028495
- [29] Mestre Â. An Algebraic Representation of Graphs and Applications to Graph Enumeration. *International Journal of Combinatorics*. 2013;**2013**:347613, 14 pages. DOI: 10.1155/2013/347613

- [30] Liao T, Wang W-Q, Huang B, Xu J. Learning laplacian matrix for smooth signals on graph. IEEE International Conference on Signal, Information and Data Processing (ICSIDP). 2019;**2019**: 1-5. DOI: 10.1109/ICSIDP47821.2019.9173468
- [31] Shi J, Cheung M, Du J, Moura JMF. Classification with vertex-based graph convolutional neural networks. 52nd Asilomar Conference on Signals, Systems, and Computers. 2018;**2018**: 752-756. DOI: 10.1109/ACSSC.2018.8645378
- [32] Tang S, Li B, Yu H. ChebNet: Efficient and Stable Constructions of Deep Neural Networks with Rectified Power Units using Chebyshev Approximations. arXiv. 2019
- [33] Xiao J, Xie Y, Wen Y. The short-time traffic flow prediction at ramp junction based on wavelet neural network. IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). 2021;**5**:664-667. DOI: 10.1109/IAEAC50856.2021.9390960
- [34] Yang H, Pan Z, Tao Q. Online learning for time series prediction of AR model with missing data. Neural Processing Letters. 2019;**50**(3): 2247-2263. DOI: 10.1007/s11063-019-10007-x
- [35] Wu Z, Pan S, Long G. Graph wave net for deep spatial-temporal graph modeling. Twenty-eighth International Joint Conference on Artificial Intelligence IJCAI. arXiv preprint arXiv: 1906.00121, 2019
- [36] Jin Y, Jia Z, Wang P, Sun Z, Wen K, Wang J. Quantitative Assessment on Truck-Related Road Risk for the Safety Control via Truck Flow Estimation of Various Types. IEEE Access. 2019;**7**: 88799-88810. DOI: 10.1109/ACCESS.2019.2924699
- [37] Jin Y, Gao Y, Wang P, Wang J, Wang L. Improved Manpower Planning Based on Traffic Flow Forecast Using a Historical Queuing Model. IEEE Access. 2019;**7**:125101-125112. DOI: 10.1109/ACCESS.2019.2933319
- [38] Yuan W, Wang P, Yang J, Meng Y. An alternative reliability method to evaluate the regional traffic congestion from GPS data obtained from floating cars. IET Smart Cities. 2021;**3**(2):79-90. DOI: 10.1049/smc2.12001