Complex & Intelligent Systems (2025) 11:158
https://doi.org/10.1007/s40747-024-01769-6

ORIGINAL ARTICLE

Short-term urban traffic forecasting in smart cities: a dynamic diffusion spatial-temporal graph convolutional network

Xiang Yin[1] · Junyang Yu[2] · Xiaoyu Duan[3] · Lei Chen[2] · Xiaoli Liang[2]

Received: 10 January 2023 / Accepted: 23 December 2024 / Published online: 22 January 2025

Abstract

Short-term traffic forecasting is an important part of intelligent transportation systems. Accurately predicting short-term traffic trends can avoid traffic congestion and plan travel routes, which is of great significance to urban management and traffic scheduling. The difficulty of short-term urban traffic forecasting is that the traffic flow is random and will be dynamically changed by the traffic conditions of nearby nodes. In order to solve this problem, this paper proposes a model based on Dynamic Diffusion Spatial-Temporal Graph Convolutional Network. It first combines the dynamic generation matrix and the static distance matrix to grasp real-time traffic conditions, and then introduces the diffusion random walk strategy to capture the correlation of spatial nodes. Finally, the convolutional LSTM module is used to mine the spatio-temporal dependence of traffic data to improve the accuracy of traffic prediction. Compared to several baseline models, the experimental results show that the model is 7% better than other models on several metrics and demonstrates the necessity of the module through ablation experiments.

Keywords Traffic forecasting · Graph convolutional network · Spatial-temporal · Dynamic generation

Introduction

With the continuous improvement of residents' living standards, the number of urban motor vehicles is also increasing. Consequently, problems such as traffic congestion, frequent accidents, and air pollution are becoming increasingly severe. Developing an integrated and efficient intelligent transportation infrastructure and promoting the integrated application of intelligent transportation based on big data and artificial intelligence represent significant policy objectives. Predicting traffic flow and vehicle speed in advance can help alleviate traffic problems through traffic control and diversion. In the development of smart cities, traffic flow is a crucial measure for short-term traffic status, and abnormalities in traffic flow can offer significant insight for traffic control [1, 2]. Hence, accurate, timely, and high-accuracy short-term traffic speed prediction can effectively alleviate traffic congestion, enhance the usage of road resources, and decrease economic losses and urban air pollution arising from traffic gridlocks.

In the field of intelligent transportation systems, it has been observed by researchers that traffic data often exhibits significant periodicity. As a result, historical data can be useful in predicting future trends. In the initial stages of this research, scholars from around the world relied primarily on linear models, such as historical average models, filter models [3, 4], and others, to forecast traffic data. The principles of these models are fairly straightforward and they offer high calculation efficiency. However, when it comes

✉ Xiang Yin
yinxiang@henu.edu.cn

Junyang Yu
10250140@henu.edu.cn

Xiaoyu Duan
327086056@qq.com

Lei Chen
chenlei@henu.edu.cn

Xiaoli Liang
liangxiaoli@henu.edu.cn

1 School of Software, Intelligent Data Processing Engineering Research Center of Henan Province, Henan University, Henan, China

2 Present address: School of Software, Henan University, Henan, China

3 School of Computer Science, Wuhan University, Wuhan, China

to traffic speed data containing vast amounts of information and intricate variations, the prediction accuracy of these models drops considerably. Some models deal with linear time-varying multiple-input multiple-output systems with non-uniform trial lengths [5]. Some researchers combine physics and mathematics with equations and graphs to solve problems [6]. Subsequently, with the advent of neural network models, some non-linear regression models such as cross-scale detection [7], Deep Belief Network (DBN) and Recurrent Neural Network (RNN) [8] have achieved impressive results. The models proposed by Du [9] and Francisco [10] present notable benefits in processing time-series data, leading to enhanced computing efficiency. However, short-term traffic data exhibits a potent temporal and spatial correlation.

In recent years, graph-based spatio-temporal traffic prediction technology has emerged as a research priority for investigating traffic data's spatio-temporal characteristics, and it has yielded outstanding outcomes [11–13]. Typically, time-series traffic data shows a degree of periodicity, particularly on weekdays when it demonstrates a comparable periodic trend. However, during short-term traffic forecasting, the occurrence of traffic accidents and peak congestion events causes traffic conditions to dynamically change over time, leading to the non-smoothness of traffic data time series. This real-time fluctuation significantly affects upstream and downstream traffic. When a node is congested, adjacent nodes will also change accordingly. In addition, it is essential to consider the intricate spatial dependencies among sensors placed on the road network, as even closely located sensors may exhibit vastly different traffic states [14]. Consequently, to achieve precise short-term traffic forecasting, the primary main focus is on the highly dynamic alterations and intricate spatio-temporal dependencies of traffic flow.

In the prediction of short-term traffic, there exists a complicated relationship between periodic patterns and dynamic irregularities in traffic data. At the macro level, the data shows similar patterns, such as homogeneous large-scale dynamic congestion during peak hours and steady deviations in the traffic pattern during weekdays and weekends. At the micro level, the traffic data displays dynamic and complex fluctuations that cause significant changes in the traffic state at any moment. There is also spatial heterogeneity, leading to traffic pattern differences in different regions or locations. This paper aims to present a short-term urban traffic prediction model based on a Dynamic Diffusion Spatio-Temporal Graph Convolutional Network (DDSTGCN) to address the dynamic change pattern of the road network. Firstly, the dynamic generation matrix is combined with the original road network matrix in order to capture the spatial correlation between traffic and real-time road conditions. Additionally, a two-way random walk approach is implemented to simulate the short-term traffic trends of future traffic data. Finally, a convolutional LSTM is utilized on top of the LSTM network to capture both the long-term dependence of the traffic data and the spatial dependence among the data. Compared with other state-of-the-art models, the model in this paper can dynamically capture the node data flow relationship, which improves the local information modeling capability and is more suitable for real-time short-term traffic prediction. The key outcomes of this paper are as follows:

1. The design of the dynamic adjacency matrix module improves the model's ability to perceive temporal changes in the road network and focus on dynamic changes in spatial nodes. The module achieves this by combining the dynamic matrix produced at every moment with the original road network matrix. Additionally, it includes global correlation between distant regions, thus reducing reliance on a priori information about the road network.
2. Design of a bi-directional random walk module for simulating short-term traffic flow trends. The module captures the dynamic spatio-temporal dependence between nodes by analyzing the correlation between the probability distributions of each node. This approach considers traffic flow as a simultaneous change over time and space, thus exploring the correlation generated by the joint action of these factors. By avoiding separate and independent modeling for time and space, this module offers a more comprehensive understanding of traffic patterns.
3. Regional traffic flow patterns exhibit evident temporal and spatial characteristics that vary across regions. Because traffic is mobile, regional traffic flow changes have a complicated relationship with obvious multi-period features. Convolutional units are employed to widen the network's sensory field and capture the impact of regional traffic flow. The closer the local nodes are, the more interdependent they are, and it is easier to enhance the dynamic correlation within the vicinity. In addition, LSTM displays a superior performance in learning short-term temporal dependencies. The model also heightens the depth of the overall structure by incorporating stacked spatio-temporal coding and decoding modules, which improve the ability to capture the proximity and periodicity of the data.

The remainder of this paper is organized as follows: Sect. Related work presents the related background work. Section Problem Statement describes the problem statement. More details of this DDSTGCN model are explained in Sect. The DDSTGCN Model. Experiments and Results are given in Sect. Experiments and Results. The conclusion can be found in Sect. Conclusion.

Related work

Classic model

For some time, researchers have focused on predicting spatiotemporal series data. Many of these techniques are founded on classic statistical methods and shallow neural networks. For instance, Williams [15] utilises Autoregressive Moving Average (ARIMA) to model traffic data. In this study, the traffic flow for a solitary variable is modeled as an autoregressive moving average process for forecasting traffic flow. Sun et al. [16] employed a Bayesian network for predicting traffic flow, which is an effective approach for analyzing the traffic flow relationship among adjacent road sections. Lippi et al. [17] provided a comprehensive overview of existing short-term traffic flow prediction techniques in the context of probabilistic graphical models. They conducted experimental comparisons and performance analysis to further investigate their effectiveness. These superficial models possess a modest structure and swift prediction speed. Nevertheless, they prove unsuitable for intricate nonlinear issues and prove challenging to implement with extensive traffic datasets.

Deep network model

In recent years, researchers have utilised deep learning to analyses spatiotemporal data. Bucur et al. [18] suggested the implementation of adaptive fuzzy neural networks for traffic prediction. In addition, they proposed an architecture to monitor the probability distributions associated with weather conditions, seasons, or other factors. It is hoped that this approach will lead to more accurate predictions of traffic patterns. Ma et al. [19] defined traffic flow prediction as an image learning problem. They applied traditional neural networks to large-scale traffic network analysis and found that artificial networks exhibit self-learning ability and can reflect real-time traffic changes after network training. Yao et al. [20] presented the Deep Multi-View Spatial-Temporal Network (DMVST-Net), a traffic flow prediction method that integrates Convolutional Neural Networks (CNN) and LSTM networks to jointly model spatiotemporal dependencies. In 2018, they suggested a spatiotemporal dynamic network that can dynamically ascertain the similarity between locations [21]. Nonetheless, these models have evident limitations on input data and are not entirely appropriate for graph-structured data.

Attention mechanism model

The emergence of the attention mechanism has opened up new ideas for people to study spatiotemporal prediction. Liang et al. proposed a multi-level attention network [22], which adjusts the correlation between multiple sensors based on encoder-decoder adaptation. Ding et al. [23] propose a multi-modal spatio-temporal graph attention network, using a multi-head attention module and two relational attention modules (i.e. intra-modal and inter-modal attention) to explicitly correlation modeling. Zhou et al. [24] proposed a graph attention learning framework based spatio-temporal predictive, which is effective in handling both explicit and implicit dynamic spatial connections and learning high-dimensional spatio-temporal features. The attention mechanism has a powerful performance in the processing tasks of sequence data, but for spatio-temporal prediction, the computational complexity is high and the memory space requirement is high.

Graph neural network

Bui [25] et al. introduced several important spatio-temporal graph neural networks and tested the contribution of key components on benchmark datasets. Min [26] proposed a Spatial-Temporal Graph Social Network, which leverage the temporal attention mechanism to capture temporal features and design a method analyzing temporal attention distribution to improve the interpretation ability. Seo et al. proposed a model known as the Graph Convolutional Recurrent Network (GCRN) to predict spatiotemporal data [27]. However, determining how to optimally combine Recurrent Neural Networks and Graph Convolutional Networks for superior model performance can be challenging. Reference [28] introduces the spatiotemporal graph convolution network (STGCN), which exhibits better computational efficiency than traditional CNN and RNN models, and can effectively learn spatiotemporal characteristics. Li et al. [29] introduced a diffusion convolutional recurrent neural network (DCRNN) consisting of an encoder and decoder network that can capture profound spatiotemporal dependency characteristics. Additionally, Kong et al. [30] presented a multimodal passenger flow prediction framework using a graph convolutional network (GCN) to extract significant topological information from the graph and implement a deep clustering method for identifying concealed mobility patterns. Most current real-time traffic prediction methods consider the relationship between space and time. However, as traffic data is subject to frequent changes, a node's alteration will affect the adjustment of its adjacent nodes accordingly. Most algorithms, however, fail to account for capturing the dynamic dependencies between adjacent traffic nodes.

Problem statement

The performance of traditional deep learning methods to deal with non-Euclidean spatial data is not satisfactory in some cases. So, the spatiotemporal prediction technology based on graph structure has been a hotspot in recent years. By mapping the nodes of the spatial-temporal data graph directly to the graph neural network structure, the complex spatial relationships can be well reflected. The spatial structure and relationship between these series data can be represented by a graph $G = (V, E, W)$ to describe the topological structure, where V is a set of the nodes, $V = \{v_1, v_2, \ldots, v_N\}$, N is the number of nodes, and E is the set of edges. $W \in R^{N\times N}$ is a weighted adjacency matrix to represent the proximity of the nodes (e.g., a function of their spatial network distance). Then the time series traffic data is represented as $X = \{X_0, X_1, \ldots, X_t, \ldots\}$, where $X \in R^{N\times P}$, P is the feature dimension, X_t represents the input value at time step t. In spatial-temporal prediction, given the graph G and its historical T steps graph inputs, find a function F to forecast the next ϵ steps data as outputs. Thus, the problem is described as the following formula (1):

$$\hat{Y} = \{X_{t+1}, X_{t+2}, \ldots, X_{t+\epsilon}\} = F\{X_t, X_{t-1}, \ldots, X_{t-T+1}; G\} \tag{1}$$

The DDSTGCN model

In this section, Fig. 1 illustrates the traffic prediction model. The DDSTGCN model comprises several components. Firstly, the model generates a dynamic matrix from the current input. This dynamic matrix is then merged with the initially generated distance matrix. Additionally, the inputs undergo training through diffusion bidirectional random walks strategy in the mixed matrix. Furthermore, obtaining temporal dependence is a crucial issue when making predictions. The DDSTGCN model utilizes the encoder-decoder framework and convolution LSTM units to capture temporal dependence. Further details about the core components of the model can be found in the following section.

Preprocess

In order to better represent the distance between traffic nodes, this paper transforms the relationship between nodes into graph structure $G = (V, E, W)$, and uses the static matrix W to represent the distance between nodes. This process of generating the static matrix is shown in Fig. 2.

Real-world sensor locations are first extracted and a graph structure is constructed, then a static matrix is built according to the pairwise road network distances between sensors in the real world. If $(v_i, v_j) \in E, W_{ij} > 0$, otherwise, if $(v_i, v_j) \notin E, W_{ij} = 0$. Inspired by the paper [31], we define the W_{ij} with the threshold Gaussian kernel.

$$W_{ij} = \exp\left(-\frac{\text{dist}(v_i, v_j)}{\sigma^2}\right), \text{dist}(v_i, v_j) \le \theta \tag{2}$$

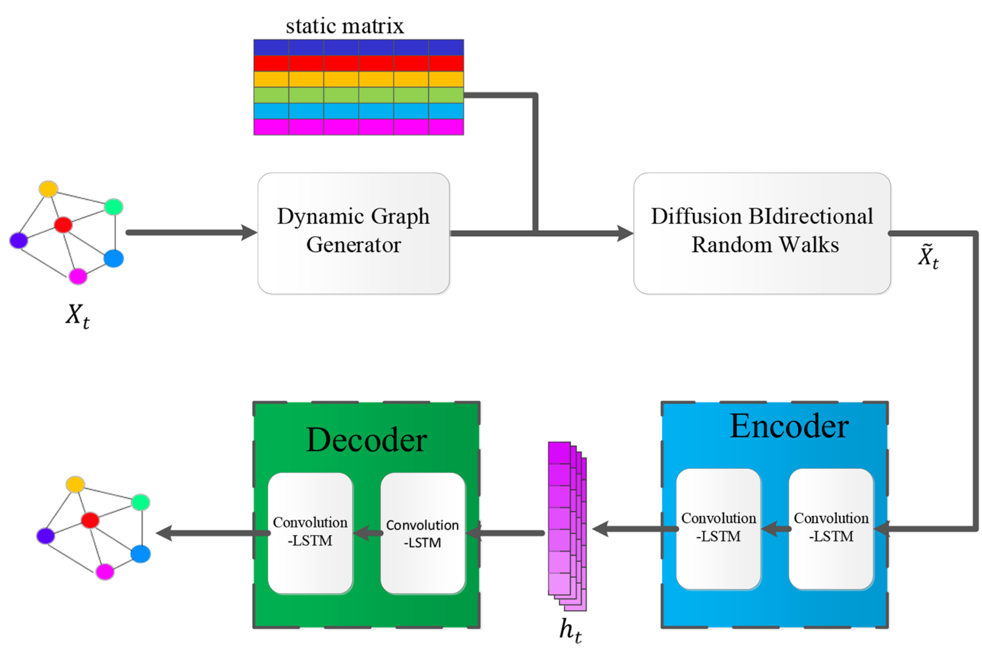


Fig. 1 The framework of dynamic diffusion spatial-temporal graph convolutional network

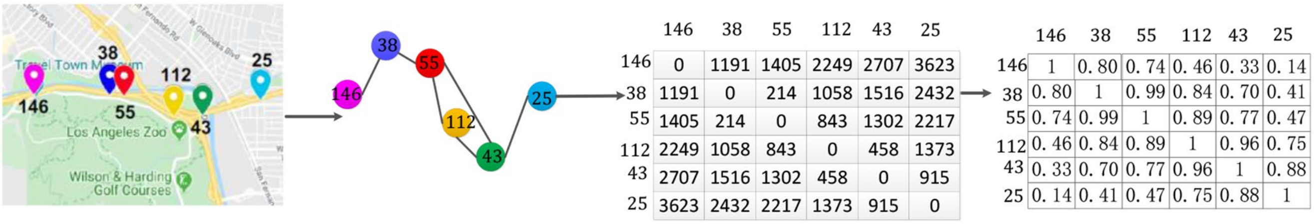
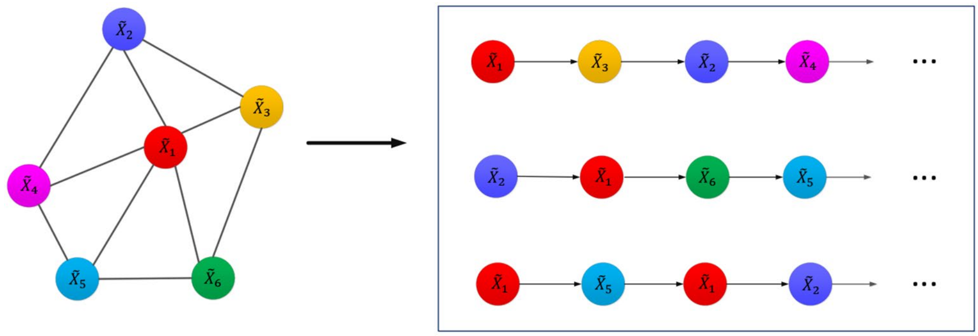


Fig. 2 The process of generating the static matrix

Fig. 3 Random walk



Where $dist(v_i, v_j)$ denotes the road distance from node v_i to node v_j. And σ is the standard deviation of distances and θ is the threshold, which Controls the distribution and sparsity of the adjacency matrix separately. Figure 2 shows the preprocessing results of the METR-LA dataset, where σ=2584.43 and θ are set to 0.1 in the METR-LA dataset, and the corresponding static matrix can be obtained after calculation.

Dynamic graph generator

To calculate the change of the entire traffic state at each time step, this paper generates a dynamic graph matrix in time step *t*. The dynamic graph generation module mainly involves combining the current traffic node input X_t, timestamp information T_t, and hidden state H_t.

$$V_t = [X_t; T_t; H_t] \tag{3}$$

Then this paper takes the dot product of the embedding vectors of the source node and the target node to obtain the dynamic adjacency matrix.

$$\boldsymbol{E_t} = \text{Relu}\left(\tanh\left(V_t V_t^T\right)\right) \tag{4}$$

$$A = E_t \odot W \tag{5}$$

Matrix A is the dynamic adjacency matrix based on semantic similarity at time t.

Diffusion bidirectional random walks

Because urban traffic data is not only periodic but also random, specifically the random fluctuations of short-term traffic conditions are greater. Such random fluctuations can easily impact the traffic conditions of surrounding nodes, specifically downstream nodes. Once adjacent nodes become congested, this traffic state may spread to other nearby nodes. Therefore, this paper simulates this scenario using the random walk strategy on the graph and converts the spatial correlation between neighboring nodes and upstream and downstream nodes into the graph structure. The random walk is a mathematical statistical model that usually denotes an undetectable or statistical pattern. In computer science, the random walk forms the basis of the diffusion process, illustrated in Fig. 3. And it has a good effect when combined with a variety of graph networks. In this paper, the diffusion process is characterized by a random walk on graph G with restart probability $\alpha \in [0,1]$. We define an out-degree diagonal matrix $D = diag\,(W1)$, and 1 denotes all one vector. After many time steps, such the Markov process converges to a stationary distribution $P \in R^{N\times N}$, illustrated as follows function (6).

$$P = \sum\nolimits_{k=0}^{\infty} \alpha(1-\alpha)^k (DA)^k \tag{6}$$

In practice, *k* is the diffusion step and is limited. Our paper also adds the reversed direction diffusion process. This bidirectional diffusion process better reflects the change of traffic flow in upstream and downstream, which can capture more spatial information. The diffusion operation on the graph is defined as the following function (7).

$$\widetilde{X}_t = \sum{}_{k=0}^{K}(\alpha\,(1-\alpha\,)^k(DA)^k + \alpha\,(1-\alpha\,)^k\left(DA^T\right)^k)X_t \tag{7}$$

Where K is a K-step truncation of the diffusion process. $\widetilde{X}_t \in R^{N\times P}$ is the output. DA and DA^T represent the transition matrices of the diffusion process and the reverse one, respectively.

Graph convolutional network module

The GCN Module consists of a set of graph convolution operations. The input of the graph convolution operation is the diffusion graph signal matrix DA, and each node aggregates the characteristics of itself and its neighbors at adjacent time steps. An aggregation function is a linear combination whose weights are equal to the weights of the edges between a node and its neighbors. Then a fully connected layer with an activation function is added to transform the features of the nodes into a new space. The graph convolution operation can be expressed as follows:

$$\mathrm{GCN}\left(h^l\right) = \sigma_g\left(DAh^{(l-1)}W + b\right) \tag{8}$$

where DA diffusion map signal matrix, $h^{(l-1)}$denotes the input of the lth layer map convolution, W and b are learnable parameters, and σ_g is the nonlinear activation function $Relu$, which spreads faster and converges faster. Convolution-LSTM is used as the main framework because of the need to aggregate spatio-temporal features of traffic data.

Convolution-LSTM

The Convolutional Long Short-Term Memory (Convolution-LSTM) network is an optimization model based on the fully connected LSTM (FCLSTM) proposed by Shi [32], which is mainly used to solve the precipitation problem. In this paper, the convolutional LSTM is introduced into traffic prediction, and its convolutional properties can be used to obtain the temporal information flow while also processing the spatial correlation in the data.

The essence of Convolution-LSTM is the same as LSTM, which takes the output of the upper layer as the input of the next layer. The difference between the two components is that after adding the convolution operation, Convolution-LSTM can not only get the temporal relationship, but also extract spatial features using the convolution layer. In this way, Convolution-LSTM can simultaneously capture temporal and spatial features (spatiotemporal features). The formulas are as follows.

$$f_t = \sigma\left(A_{xf} * \tilde{X}_t + GCN_f\left(H_{t-1}\right) + A_{cf} \odot C_{t-1} + b_f\right) \tag{9}$$

$$I_t = \sigma\left(A_{xI} * \tilde{X}_t + GCN_I\left(H_{t-1}\right) + A_{cI} \odot C_{t-1} + b_I\right) \tag{10}$$

$$C_t = f_t \odot C_{t-1} + I_t \tanh\left(A_{xc} * \tilde{X}_t + \mathrm{GCN}_c\left(H_{t-1}\right) + b_c\right) \tag{11}$$

$$o_t = \sigma\left(A_{xo} * \tilde{X}_t + GCN_0\left(H_{t-1}\right) + A_{co} \odot C_t + b_o\right) \tag{12}$$

$$H_t = o_t \odot \tanh\left(C_t\right) \tag{13}$$

Where $*$ represents the convolution operator and $\odot$ represents the Hadamard product. σ is the sigmoid function sigmoid, which is a smoothly differentiable step function, and it stands for information retained or discarded. The $\tanh$ activation function has a larger amplitude, which alleviates the problem of gradient disappearance to a certain extent. C_t denotes the cell outputs and H_t denotes the hidden states. A full connect layer is used to process the hidden state of the convolution-LSTM component at timestamp t, and the result is denoted by $\widehat{Y}$.

$$GCN\left(h^l\right) = \sigma_g\left(DAh^{(l-1)}W + b\right)\#(8) \tag{14}$$

The parameters W_y and b_w map the concatenation to the size of the decoder hidden states. This framework employs the Encoder-Decoder architecture. In order to capture the temporal dependency of the historical time series, two-layers Convolution-LSTM networks are used in both the Encoder and Decoder networks.

Optimization strategy

In this paper, the Adam optimizer [33] is used during the training phase. The Adam optimizer not only uses momentum as the parameter update direction, but also can adaptively adjust the learning rate to improve gradient descent. Because this method is smooth and differentiable, it can be trained via a back-propagation algorithm by minimizing the mean squared error between the predicted vector $\widehat{y}_T$ and the true vector y_T at time step t as given in (15).

$$o\left(\widehat{y}_T\,, y_T\,\right) = \frac{1}{N}\sum{}_{i=1}^{N}\left(\widehat{y}_T^i - y_T^i\right)^2 \tag{15}$$

Where N is the number of training samples.

Experiments and results

In this section, we conduct extensive experiments using five models on three real-world spatial-temporal datasets for forecasting tasks. Additionally, we introduce the parameter settings for the DDSTGCN model and the corresponding evaluation metrics. Furthermore, we validate and interpret the DDSTGCN model. All experiments are compiled and tested on a Linux cluster (CPU: Intel(R) Xeon(R) CPU E5-2620 v2 @ 2.10 GHz, GPU: NVIDIA Titan X GPU).

Data sets

METR-LA

The METR-LA dataset from the Los Angeles Metropolitan Transportation Authority contains average traffic speed measured by 207 loop detectors on the highways of Los Angeles County ranging from Mar 2012 to Jun 2012.

PEMS-BAY

The PEMS-BAY dataset from California Transportation Agencies (CalTrans) contains average traffic speed measured by 325 sensors in the Bay Area ranging from Jan 2017 to May 2017.

PEMSD4

The PeMSD4 dataset refers to the traffic flow data in the San Francisco Bay Area. There are 307 loop detectors selected within the period from 1/Jan/2018 to 28/Feb/2018.

These three data sets are divided into a training set (60%), a validation set (20%), and a test set (20%) in chronological order. The benchmark data set statistics are summarised in Table 1. To ensure each feature makes an equal contribution to the results, the data has been preprocessed using a normalisation method.

Evaluation metrics

In order to compare the effectiveness of various time series prediction models, this paper uses three corresponding performance indices to evaluate these models. These measures enable comparison between predicted and actual values. The three performance measures are listed below:

(1) root squared error (RMSE) [34].

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum\nolimits_{i=1}^{N}(y_t^i - \widehat{y}_t^i)^2} \tag{16}$$

(2) mean absolute error (MAE).

$$\mathrm{MAE} = \frac{1}{N}\sum\nolimits_{i=1}^{N}|y_t^i - \widehat{y}_t^i| \tag{17}$$

(3) mean absolute percentage error (MAPE).

$$\mathrm{MAPE} = \frac{1}{N}\sum\nolimits_{i=1}^{N}\frac{|y_t^i - \widehat{y}_t^i|}{y_t^i} \tag{18}$$

Where y_t is the true target at time t and $\widehat{y}_t$ is the predicted value at time t. All three indices are widely used in regression tasks.

Baseline models

LSTM [35]: The LSTM model is a variant algorithm of RNN that overcomes the limitation of vanishing gradients in RNNs. Since it can capture long-term dependence, it has achieved good results in many time series tasks.

STGCN [28]: The paper introduces strategies to efficiently model the temporal dynamics and spatial dependence of traffic flows. To fully exploit the spatial information, the traffic network is modeled using generalized graphs instead of viewing the traffic flow as individual discrete parts (e.g., lattices or fragments). To deal with the shortcomings of recurrent neural networks, a full convolutional structure is deployed on the time axis to accelerate the training process of the model. The paper proposes a neural network architecture consisting of multiple spatio-temporal graph convolutional blocks (a spatio-temporal graph convolutional network) to predict traffic conditions.

DCRNN [29]: A Diffusion convolution recurrent neural network, which combines graph convolution networks with recurrent neural networks in an encoder-decoder manner. The traffic flow is first modeled as a diffusion process, using a diffusion convolution operation to capture the spatial dependence, and then a code-and-decode structure with scheduled sampling to capture the temporal dependence.

GraphWaveNet(GWN) [36]: The Graph WaveNet framework graph employs convolutional layers to capture spatial dependencies, while learning the adjacency matrix

Table 1 Summary statistics

Datasets	Samples	Nodes	Edges	Sample Rate	Input length	Output length
METR-LA	34,272	207	1515	5 min	12	12
PEMS-BAY	52,116	325	2369	5 min	12	12
PEMSD4	16,992	307	340	5 min	12	12

Table 2 Performance comparison of different models for forecasting

	T	Metric	LSTM	STGCN	DCRNN	GWN	MTGNN	PSOBILSTM	DDSTGCN
METR-LA	5 min	MAE	4.1549	2.5936	2.4871	2.3519	2.4814	3.6588	**2.2655**
		RMSE	10.1753	4.7661	4.1354	4.2367	4.3142	4.9152	**3.9165**
		MAPE	22.3255	6.3315	5.6650	5.9985	6.0517	5.8874	**5.5463**
	10 min	MAE	8.2963	2.8311	2.7351	**2.5654**	2.7279	2.9130	2.5824
		RMSE	14.4320	6.0279	5.3216	5.4541	5.4159	5.7441	**4.8837**
		MAPE	27.9889	7.8848	6.6256	6.9712	**6.4859**	8.0369	6.5762
	15 min	MAE	10.7354	3.2817	2.7961	2.9865	3.1616	2.8487	**2.6721**
		RMSE	16.3813	6.1582	5.3919	5.8912	6.0433	5.7745	**5.2004**
		MAPE	30.5195	8.9471	7.2332	7.8868	8.5948	7.3692	**6.9392**
PEMS-BAY	5 min	MAE	2.2347	1.1318	1.0872	1.0928	0.9874	1.2311	**0.9007**
		RMSE	4.4221	2.0599	1.6467	1.7451	**1.5348**	1.6595	1.6214
		MAPE	11.2319	1.8571	1.8481	1.7773	1.6968	1.8738	**1.6533**
	10 min	MAE	3.5629	1.3257	**1.1247**	1.2764	1.2346	1.4669	1.1357
		RMSE	6.1255	2.8414	2.4341	2.3187	2.26914	2.3762	**2.2579**
		MAPE	12.4876	2.6447	2.4411	**2.2978**	2.4599	2.6439	2.3268
	15 min	MAE	6.5413	1.4262	1.3549	1.3168	1.4059	1.4117	**1.3137**
		RMSE	9.9325	3.0715	2.7676	2.6959	2.9296	2.9888	**2.6354**
		MAPE	14.2815	2.8857	2.6439	2.5853	2.7370	2.6417	**2.5537**
PEMSD4	5 min	MAE	18.9541	18.3217	**16.3928**	17.1811	16.8471	18.1134	16.4381
		RMSE	37.2329	29.3174	27.9524	**26.9985**	28.6498	30.6616	27.7554
		MAPE	15.5372	13.5184	**10.4396**	11.0409	12.3311	13.9747	10.4571
	10 min	MAE	22.2377	21.0487	17.1523	**16.8733**	17.7182	20.3377	16.9415
		RMSE	42.3463	31.3341	29.1167	29.0415	29.0615	33.6961	**28.6779**
		MAPE	19.5563	14.4744	12.6958	12.9755	13.5958	13.4876	**12.4741**
	15 min	MAE	24.3591	22.1674	19.4864	18.0532	18.7570	19.9930	**17.9544**
		RMSE	48.6917	33.4559	28.9102	29.6380	29.8551	31.4544	**29.2763**
		MAPE	22.4419	15.6975	13.9418	14.2761	15.9035	14.1918	**13.6217**

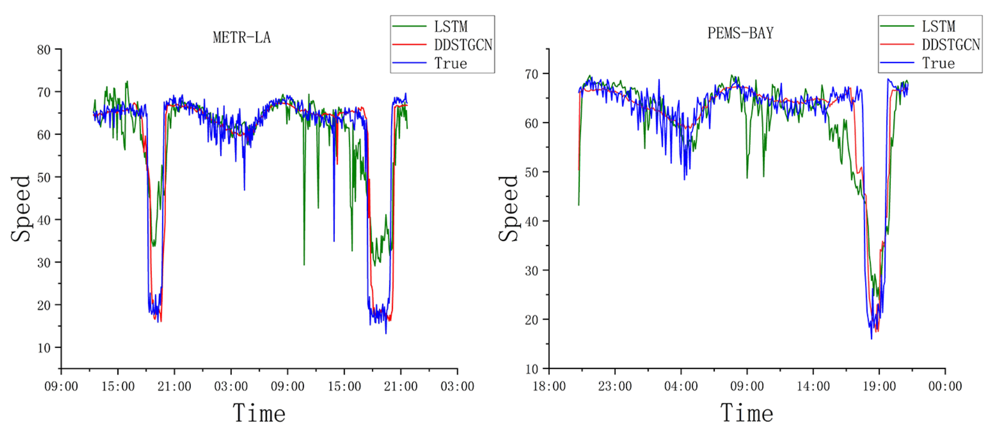


Fig. 4 Time series forecasting visualization

from the data through end-to-end training and employing stacked dilated casual convolution to capture temporal dependencies.

MTGNN [37]: Firstly, the MTGNN model automatically extracts the uni-directed relations among variables through a graph learning module, into which external knowledge like variable attributes can be easily integrated. Then the graph convolution module and the temporal convolution module capture the spatio-temporal dependencies between time series, and in addition, the three models can be trained jointly end-to-end.

PSO-BILSTM [38]: This model adopts the PSO method to search the best parameters of the model globally, considers replacing linear weights with nonlinear variable inertia weights, and adopts the Bi-LSTM model to effectively capture the time dependence and patterns in traffic flow data.

Performance comparison

In this experiment, the DDSTGCN model parameters are set as follows: θ is 0.1, timestep is 10, K-step is 3, and LSTM

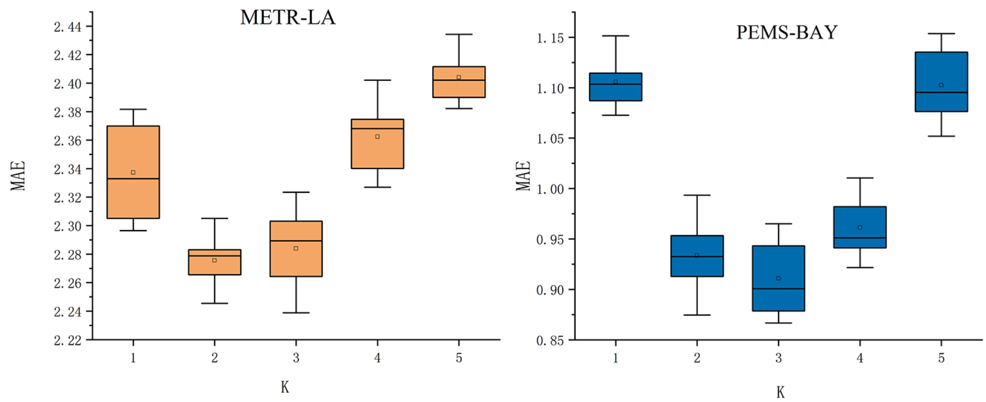


Fig. 5 MAE vs. Step of diffusion process over METR-LA (left) and PEMS-BAY (right)

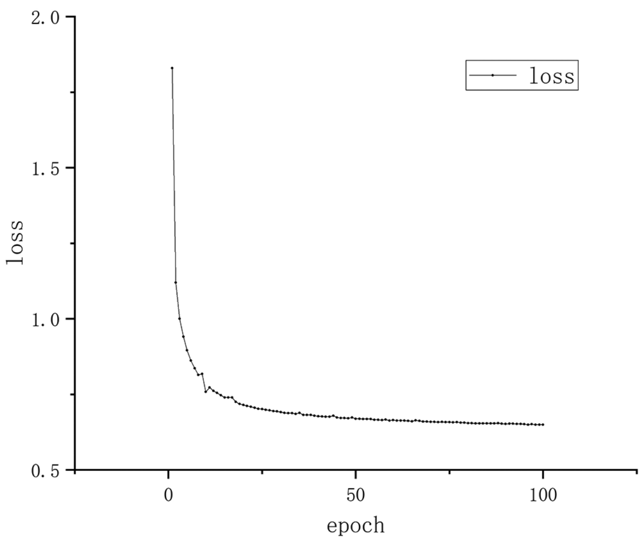


Fig. 6 Train loss on METR-LA dataset

units are 64,256. Subsequent verification will be carried out for some of the experimental parameters.

The DDSTGCN model is compared to six other baseline models across three real-world datasets. The results are presented in Table 2, with the best outcome highlighted in bold. Our model's better performance with statistical significance across all three evaluation metrics is evident. Certain phenomena are evident within these datasets; (1) when compared to LSTM models, other models can better capture the spatial features of the road, resulting in enhanced performance. (2) Although the STGCN model employs a graph convolutional network to represent the road network, it fails to capture its global spatial information. (3) The PSOBILSTM model applies nonlinear variable inertia weights to the PSO technique, although it improves the convergence speed and can effectively capture time dependencies and patterns in traffic flow data. However, BILSTM does not fully exploit the spatial properties of traffic data, especially the dynamic changes. (4) The MTGNN model solely employs nodes as exogenous series, neglecting the distance relationship among them. Therefore, the MAE metric of the DDSTGCN model is 7.9% higher than that of the MTGNN model. (5) With the dynamic graph generator component, the DDSTGCN model has improved in capturing the real-time traffic status. Therefore, it can be concluded that the DDSTGCN model is superior to the DCRNN model, as demonstrated by a 6.4% lower MAE. (6) In contrast, the Graph WaveNet model utilizes a stacked spatial-temporal layer, featuring separate GCN layers with distinct parameters that cater to a wider range of temporal inputs. Nevertheless, the model is susceptible to losing local information when the expansion rate increases. The random walk strategy of the DDSTGCN model can depict the condition of neighboring traffic nodes and disperse node data to adjacent nodes promptly. Hence, the DDSTGCN model outperforms the Graph WaveNet model, particularly on more intricate datasets like METR-LA. Specifically, the DDSTGCN model shows a 6% improvement over the GWN model. The DDSTGCN mode surpasses the GWN model in most metrics, which highlights the effects of dynamic spatiotemporal dependency modelling.

To enhance comprehension of the model, Fig. 4 portrays the predicted and actual 15-minute-ahead values. Granting discretion regarding subjective evaluation, DDSTGCN displays more stable and less fluctuating predicted values than the LSTM model across various datasets. At 18:00–20:00, the actual value underwent abrupt changes owing to traffic congestion from commuting to work. The DDSTGCN model is capable of predicting the initiation and conclusion of low peak periods with greater precision and speed in response to peak flow dynamics than existing techniques. This is attributable to the model's superior ability to model spatio-temporal correlations and identify alterations in proximity sensors, which has resulted in a substantial improvement in prediction outcomes.

To assess the efficacy and sensitivity of the Diffusion random walk technique, this study examines the impact of K-step (ranging from 1 to 5) on predictive outcomes. We tested each parameter 10 times and plotted box and whiskers diagrams (shown in Fig. 5) to demonstrate the effect

Table 3 Results of a Wilcoxon ranks test for DDSTGCN vs Other baseline

	p-value		*p*-value
LSTM	6.6390E-09	GWN	0.0198859
STGCN	4.6336E-13	MTGNN	4.7794E-08
DCRNN	0.0013069	PSOBILSTM	0.00036854

of different parameters. The results indicate that prediction outcomes vary with changes to the K-step in the diffusion process. It is thus advisable to establish an appropriate numerical value to enhance the accuracy of predictions. The model's prediction result is superior when K is equal to 2 or 3. Broadening the spatial dependency by increasing K comes at the expense of heightened learning complexity. Employing precise K values strikes a balance between capturing spatial dependencies and learning complexity. If the step size of K is too long, the diffusion node may be distant from the original node, which results in reduced prediction accuracy.

To verify the performance of the activation function and the optimizer, we observed the convergence of the training loss (shown in Fig. 6). Globally, the model converged quickly and smoothly, resulting in a good fit to the data.

Table 3 Presents the statistical results of the Wilcoxon signed rank test [39], which serves as a nonparametric test that detects whether there is a significant difference between the DDSTGCN and each of the compared algorithms. It is clear that there is a significant difference between the algorithm in this paper and the other baseline algorithms when a=0.05

Ablation study

In order to provide additional evidence of the efficacy of our framework, we conducted a series of ablation experiments, where we removed specific components for comparison. These experiments demonstrate the effectiveness of each individual component. Initially, we assigned names to the frameworks lacking certain components.

DDSTGCNw/oDG: The DDSTGCN model without dynamic graph generator component.

DDSTGCNw/oDW: The DDSTGCN model without the diffusion bidirectional random walks component.

DDSTGCNw/oC: Replace the convolution LSTM component with the classical LSTM component in the DDSTGCN model.

The test employs identical parameters and presents the comparative outcomes in Fig. 7. The following observations can be made: (1) Each constituent enhances the prediction precision to some extent. (2) The proximate nodes are more interdependent. Hence, removing the diffusion bidirectional random walks module has a noteworthy impact on the prediction. The dynamic graph generator component offers insights into the surrounding traffic nodes in real-time, thereby facilitating accurate prediction. The convolution LSTM component, when devoid of convolutional operation, exerts a minimal impact on the overall framework, since the LSTM network is adept at capturing long-term dependencies.

Conclusion

In this study, a short term prediction model for urban traffic (DDSTGCN) is proposed to predict the traffic condition in advance and reduce traffic congestion. The research addresses the limitation of current graph neural network-based prediction models, which do not consider the dynamic changes of road network nodes. To address this issue, we employ a dynamic neighbor matrix module to extract spatio-temporal sensing capability that does not rely on a predefined static neighbor matrix. This approach effectively enhances the representation of internal dynamic correlation attributes among road network nodes effectively. The experiments also verify the impact of dynamic node changes on the prediction. Moreover, as each city node is influenced by its neighboring nodes, the diffusive bidirectional random walk strategy can better capture the local correlation

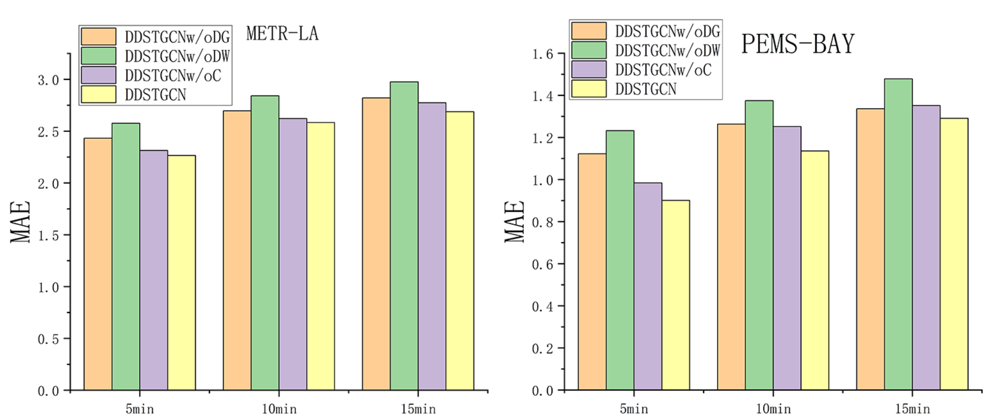


Fig. 7 Ablation study

between neighboring nodes, thereby enhancing the prediction accuracy. Additionally, the convolutional LSTM unit has the potential to improve the understanding of dynamic spatio-temporal dependencies in time series data. This paper presents comparative experiments that demonstrate the ability of the DDSTGCN-based traffic condition prediction model to significantly enhance short-term traffic condition prediction accuracy. In the future, we will focus on seeking improvements in the graph storage structure, especially when roads are subject to traffic control, resulting in dynamic changes in the graph structure, so that the model can handle more complex road situations.

Author contributions The first author constructed the scheme and wrote the manuscript. The second author reviewed the manuscript and checked the validity of the scheme. She also proofread the manuscript and corrected the grammar mistakes. The third author and the fourth author joined the discussion of the work. All authors read and approved the manuscript.

Declarations

Competing interests The authors declare that they have no competing interests.

References

1. Du WD, Zhang QY, Chen YP et al (2021) An urban short-term traffc flow prediction model based on wavelet neural network with improved whale optimization algorithm[J]. Sustainable Cities Soc 69:102858
2. Kong X, Chen Q, Hou M et al (2022) RMGen: a TriLayer Vehicular Trajectory Data Generation Model Exploring Urban Region Division and mobility Pattern[J]. IEEE Trans Veh Technol 71(9):9225–9238
3. Guo J, Huang W, Williams BM (2014) Adaptive kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification[J]. Transp Res Part C Emerg Technol 43:50–64
4. Tao HF, Qiu J, Chen YY et al (2023) Unsupervised cross-domain rolling bearing fault diagnosis based on time-frequency information fusion[J]. J Franklin Inst 360(2):1454–1477
5. Zhuang ZH, Tao HF, Chen YY et al (2022) An optimal iterative Learning Control Approach for Linear Systems with Nonuniform Trial Lengths under Input Constraints[J]. IEEE Trans Syst Man Cybernetics 53(6):3461–3473
6. Molla RR (2014) Software and Graphical Approach for Understanding Friction on a Body[C]. *Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA).* (2): 637–644
7. Shen LZ, Tao HF, Ni YZ et al (2023) Improved YOLOv3 model with feature map cropping for multi-scale road object detection[J]. Meas Sci Technol 34(4):34045406
8. Jonathan M, John FR, Rocco Z (2018) An evaluation of HTM and LSTM for short-term arterial traffic Flow Prediction[J]. IEEE Trans Intell Transp Syst. 1–3
9. Du L, Gao RB, Suganthan PN et al (2022) Bayesian optimization based dynamic ensemble for time series forecasting. Inf Sci. (591): 155–175
10. Francisco M, Francisco C, María PF et al (2022) Strategies for time series forecasting with generalized regression neural networks[J]. Neurocomputing. (491):509–521
11. Arzoo M, Neeraj K (2019) Deep learning models for traffic flow prediction in autonomous vehicles: a review, solutions, and challenges[J]. Veh Commun 20:100184
12. Kong X, Zhu B, Shen G et al (2021) Spatial-temporal-cost combination based Taxi driving Fraud Detection for Collaborative Internet of Vehicles[J]. IEEE Trans Industr Inf 18(5):3426–3436
13. Guo S, Lin Y, Feng N, Song C et al (2019) Attention based spatial-temporal graph convolutional networks for traffic flow forecasting[C]. *Proceedings of the AAAI conference on artificial intelligence*. (33):922–929
14. Moria U, Mendiburub A, Alvarezc M (2014) A review of travel time estimation and forecasting for advanced traveler information systems[J]. Transportmetrica A: Transp Sci 11(2):119–157
15. Williams BM, Hoel LA (2003) Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: theoretical basis and empirical results[J]. J Transp Eng 129(6):664–672
16. Sun SL, Zhang CH, Yu GQ (2006) A bayesian network approach to traffic flow forecasting[J]. IEEE Trans Intell Transp Syst 7(1):124–132
17. Lippi M, Bertini M, Frasconi P (2013) Short-term traffic flow forecasting: an experimental comparison of time-series analysis and supervised learning[J]. IEEE Trans Intell Transp Syst 14(2):871–882
18. Bucur L, Florea A, Petrescu BS (2010) An adaptive fuzzy neural network for traffic prediction[C]. 18th Mediterranean Conference on Control and Automation. 1092–1096
19. Ma XL, Zhuang D, He ZB et al (2017) Learning traffic as images: a deep convolutional neural network for large-scale Transportation Network Speed Prediction. Sensors 17(4):818–821
20. Yao H, Wu F, Ke J et al (2018) Deep Multi-View Spatial-Temporal Network for Taxi Demand Prediction[C]. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2588–2595
21. Yao H, Tang X, Wei H et al (2018) Modeling Spatial-Temporal Dynamics for Traffic Prediction. https://arxiv.org/pdf/1803.01254.pdf
22. Liang YX, Ke SY, Zhang JB (2018) GeoMAN: Multi-level Attention Networks for Geo-sensory Time Series Prediction[C]. Twenty-Seventh International Joint Conference on Artificial Intelligence. 3428–3434
23. Ding CY, Sun SL, Zhao J (2023) MST-GAT: a multimodal spatial–temporal graph attention network for time series anomaly detection[J]. Inform Fusion. (89): 527–536
24. Zhou BJ, Dong YX, Yang JH et al (2023) A graph-attention based spatial-temporal learning framework for tourism demand forecasting[J]. Knowledge-Based Systems. (263):110275

25. Bui KHN, Cho J, Yi H (2022) Spatial-temporal graph neural network for traffic forecasting: an overview and open research issues. Appl Intell 52(3):2763–2774
26. Min S, Gao Z, Peng J et al (2021) STGSN—A spatial-temporal graph neural network framework for time-evolving social networks. Knowl Based Syst 214:106746
27. Seo Y, Defferrard M, Vandergheynst PBX (2018) Structured Sequence Modeling with Graph[C] Convolutional Recurrent Networks. International Conference on Neural Information Processing:362–373
28. Yu B, Yin HT, Zhu ZX (2018) Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting[C]. Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence. 3634–3640
29. Li YG, Yu R, Cyrus S et al (2018) Diffusion convolutional recurrent neural network: Data-driven traffic forecasting[C]. International Conference on Learning presentation. 1–16
30. Kong XJ, Wang KL, Hou ML et al (2022) Exploring human mobility for multi-pattern passenger prediction: a Graph Learning Framework[J]. IEEE Trans Intell Transp Syst 23(9):16148–16160
31. David IS, Sunil KN, Pascal F et al (2013) The emerging field of signal processing on graphs: extending high-dimensional data analysis to networks and other irregular domains[J]. IEEE Signal Process Mag 30(3):83–98
32. Shi XJ, Chen ZR, Wang H et al (2015) Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting[C]. 28th International Conference on Neural Information Processing Systems. 802–810
33. Diederik PK, Jimmy B (2015) Adam: A method for stochastic optimization. https://arxiv.org/pdf/1412.6980v8.pdf
34. Mark P, Garrison C, Halbert W (1996) Experience with selecting exemplars from clean data[J]. Neural Netw 9(2):273–294
35. Sepp H, Jürgen S (1997) Long short-term Memory[J]. Neural Comput 9(8):1735–1780
36. Wu ZH, Pan SR, Long GD et al (2019) Graph WaveNet for Deep Spatial-Temporal Graph Modeling[C]. the 28th International Joint Conference on Artificial Intelligence. 1907–1913
37. Wu ZH, Pan SR, Long GD et al (2020) Connecting the Dots: Multivariate Time Series Forecasting with Graph Neural Networks[C]. the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 753–763
38. Bharti PR, Kranti K (2023) Short-term traffic flow prediction based on optimized deep learning neural network: PSO-Bi-LSTM[J]. Physica A 625:129001
39. Derrac J, Salvador G, Daniel M et al (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms[J]. Swarm Evol Comput. (1):3–18