This is a repository copy of *Spatial-temporal correlation graph convolutional networks for traffic forecasting*.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/195401/

Version: Published Version

Article:
Huang, R. orcid.org/0000-0001-7545-0987, Chen, Z., Zhai, G. et al. (2 more authors) (2023) Spatial-temporal correlation graph convolutional networks for traffic forecasting. IET Intelligent Transport Systems. ISSN 1751-956X

https://doi.org/10.1049/itr2.12330

Reuse
This article is distributed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivs (CC BY-NC-ND) licence. This licence only allows you to download this work and share it with others as long as you credit the authors, but you can't change the article in any way or use it commercially. More information and the full terms of the licence here: https://creativecommons.org/licenses/

Takedown
If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk/

Received: 9 November 2021 | Revised: 11 December 2022 | Accepted: 19 December 2022

DOI: 10.1049/itr2.12330

IET Intelligent Transport Systems

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH

Spatial-temporal correlation graph convolutional networks for traffic forecasting

Ru Huang[1] | Zijian Chen[1] | Guangtao Zhai[2] | Jianhua He[3] | Xiaoli Chu[4]

[1]School of Information Science & Engineering, East China University of Science and Technology, Shanghai, China

[2]Institute of Image Communication and Information Processing, Shanghai Jiao Tong University, Shanghai, China

[3]School of Computer Science and Electronic Engineering, University of Essex, Colchester, UK

[4]Department of Electronic and Electrical Engineering, University of Sheffield, Sheffield, UK

Correspondence
Ru Huang, School of Information Science, and Engineering, East China University of Science and Technology, Shanghai 200237, China.
Email: huangrabbit@ecust.edu.cn

Funding information
Natural Science Foundation of Shanghai, Grant/Award Number: 20ZR1413800; Marie Sklodowska-Curie Actions, Grant/Award Numbers: 101022280, 824019; National Natural Science Foundation of China, Grant/Award Numbers: 61673178, 61922063

Abstract

Traffic forecasting, as a fundamental and challenging problem of intelligent transportation systems (ITS), has always been the focus of researchers. Nevertheless, accurate traffic forecasting still exists some problems due to the complex spatial-temporal dependencies and irregularities of traffic flows. Most of the existing methods typically use the spatial adjacency matrix and complicated mechanism to model spatial-temporal relationships separately, while ignoring the latent spatial-temporal correlations. In this paper, a novel architecture is proposed named spatial-temporal correlation graph convolutional networks (STCGCN) for traffic prediction. First, an informative fused graph structure is constructed to better learn the complex spatial-temporal correlations, which breaks the limitation that the general spatial adjacency matrix cannot reflect temporal correlations. Moreover, spatial-temporal correlation graph convolution and gated temporal convolution are performed in parallel and they are integrated into a unified layer, which enables capturing both local and global spatial-temporal dependencies simultaneously. By stacking multiple layers, STCGCN can learn more long-range spatial-temporal dependencies. Experimental results on five public traffic datasets demonstrate the effectiveness and robustness of the proposed STCGCN in urban traffic forecasting.

1 | INTRODUCTION

With the increasing urbanization and the number of vehicles on the roads, traffic congestion is worsening and can cause significant economic loss and air pollution. Therefore, building intelligent transport systems (ITS) with more efficient management of road traffic is essential to reducing congestion and achieving greener transportation. Accurate and timely traffic flow (e.g. speed or volume) prediction cannot only effectively increase road capacity and alleviate traffic congestion but also provide traffic route planning for urban travelers to make better decisions.

Over the past few decades, significant research efforts have been devoted to traffic flow forecasting problems. Early statistical methods usually take historical traffic flow data with linear and stationarity assumptions to predict possible future traffic changes, while ignoring the non-linearity and dynamics in traffic data and thus leading to poor performance in practical applications. Traditional machine learning such as support vector regression (SVR) [1] and *k-nearest* neighbors [2] can extract the non-linearity of traffic data but still fail to reach the requirement of accurate traffic flow prediction due to the shallow structure and artificial feature selection. Recently, with the development of graph representation learning, there has arisen a new paradigm for modeling transportation networks. A series of approaches represented by graph neural networks (GNNs) [3–5] have shown their superiority in handling traffic analysis applications and are especially suitable for traffic flow prediction tasks. For example, Yu et al. [6] first formulate the problem on graphs and build a novel spatial-temporal deep learning framework with complete convolutional structures. Zhao et al. [7] combine the graph convolutional network (GCN) and the

This is an open access article under the terms of the Creative Commons Attribution-NonCommercial-NoDerivs License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.
© 2023 The Authors. *IET Intelligent Transport Systems* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology.

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

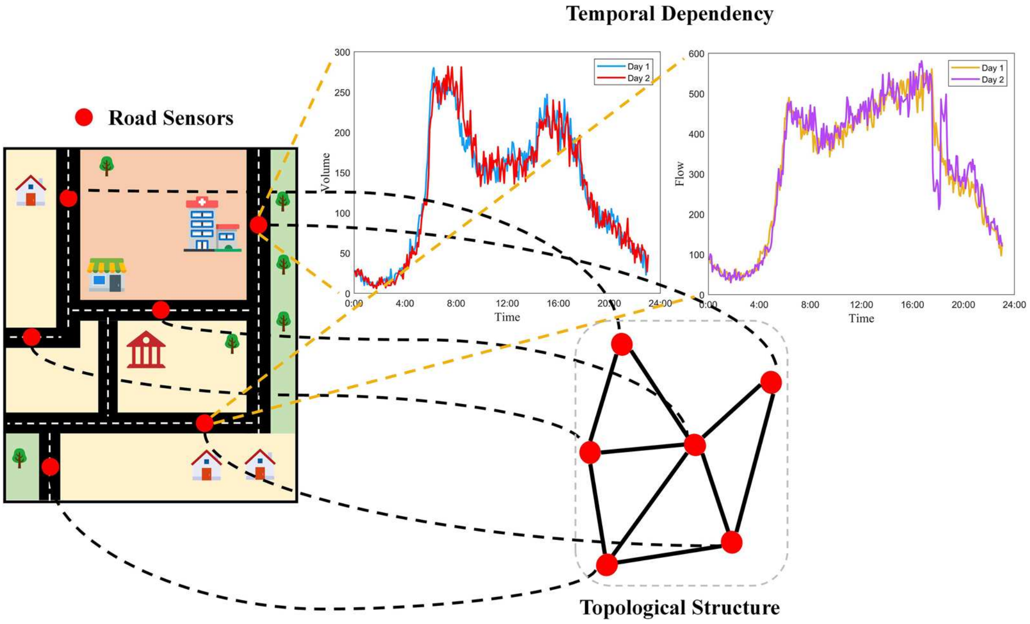


FIGURE 1 Illustration of complex spatial-temporal correlations

gated recurrent unit (GRU) to learn complex topological structures and dynamic changes for capturing spatial and temporal dependencies, respectively. Li et al. [8] construct a graph attention network, which utilizes the attention mechanism to extract both spatial information among road segments and the varying temporal dynamic features. Zheng et al. [9] present an encoder-decoder architecture to model the spatio-temporal factors on traffic conditions and distill complex correlations through an attention-based fusion operation.

Although the aforementioned methods perform well in incorporating graph structure into spatial-temporal data prediction models, these methods still exist several shortcomings. First, the majority of spatial-temporal modeling methods lack the construction of informative graphs. As shown in Figure 1, those sensor nodes in different locations may have certain correlations. In other words, traffic flow in different spatial regions would share similar variation tendencies, which are largely affected by the surroundings (e.g. office buildings, hospitals, schools etc.). Besides, it can be observed that the traffic volume exhibits similar patterns as it changes from day to day, which increases rapidly in the morning peak while dropping drastically after the evening peak. Most existing methods [10–12] directly take the given spatial adjacency matrix for graph modeling while ignoring the latent temporal similarity. For sure, researchers have already made some attempts to enrich the representation of graphs. Chen et al. [13] introduce the line graph and propose two types of edge interaction patterns to capture stream connectivity and relationship. Song et al. [14] present a spatial-temporal synchronous graph, where the localized adjacency matrix contains necessary spatial and temporal information. However, these models are either not fully considered the correlations between spatial and temporal information or fuse the spatial-temporal features properly.

Furthermore, current research on spatial-temporal data forecasting shows limited capabilities in capturing the correlations between local and global features [15, 16]. The local and global features here refer to the spatial information of nearby and distant nodes, respectively. Commonly used convolutional neural network (CNN)-based methods and message passing-based GNN methods need to stack layers for better global information extraction, which can cause over-smoothing problems. Besides, enlarging the reception field of global information by increasing the dilation rate could result in the loss of local information [17]. As a result, it is necessary to develop effective traffic flow prediction methods to ameliorate such problems.

In this paper, we propose a novel architecture, *Spatial-Temporal Correlation Graph Convolutional Network* (STCGCN) to tackle the problems mentioned above and thus improve the prediction accuracy of urban traffic flow. Motivated by approximation entropy [18], we propose an optimized cross-approximation entropy algorithm to extract the temporal correlations by measuring the similarity and irregularity between sequences. To integrate the spatial and temporal information together, a spatial-temporal correlation graph structure is proposed to aggregate the spatial and temporal features. Then, we construct a spatial-temporal correlation graph convolutional module to obtain hidden spatial-temporal dependencies. Moreover, a gated temporal convolution with dilated operation is introduced to capture long-range dependencies.

In summary, the contributions of this work are as follows:

- We construct a novel informative graph structure, namely, the spatial-temporal correlation graph, which contains abundant hidden spatial-temporal dependencies. This fusion graph is able to connect the spatial and temporal information

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

and present spatial-temporal correlations more appropriately than the spatial adjacency graph alone.

- We propose an effective architecture to capture both local and global spatial-temporal correlations simultaneously, which consists of multiple spatial-temporal correlation graph convolutional modules and a gated temporal convolution module in parallel.
- Extensive experiments on five real-world datasets are conducted to make thorough comparisons and test the robustness of the proposed model. The results show that our STCGCN outperforms baselines on traffic forecasting tasks, indicating the effectiveness of handling complicated traffic characteristics and capturing latent spatial and temporal correlations.

The remainder of this paper is organized as follows. Section 2 provides some surveys in the traffic flow prediction domain. Section 3 introduces some necessary definitions that will be used in this paper. Section 4 gives a detailed illustration for the proposed traffic flow prediction method. Extensive experimental results and analysis are presented in Section 5. Section 6 concludes this paper with future work.

2 | LITERATURE REVIEW

In this section, we briefly review the literature work. The traffic flow prediction approaches have undergone several stages, which can be roughly divided into two categories, that are, traditional parametric analysis and deep learning-based traffic forecasting.

Traditional traffic forecasting methods mainly depend on mathematical statistics or historical observations to predict future traffic states, which are limited by the capability of capturing non-linearity and merely leveraging temporal information. Early methods including historical average (HA) [19], autoregressive integrated moving average (ARIMA), and its variants [20, 21] are all based on the time series stability assumption, using the relationship between current and historical data to model the future traffic trends, and thus can hardly handle the data missing scenarios. Vector autoregressive (VAR)-based models [22] can capture pairwise relationships among all flows, but are computationally inefficient due to a large number of parameters. Afterward, machine-learning-based traffic forecasting models appeared. The *k-nearest* neighbor (KNN) algorithm [23], support vector machine (SVM) [24] are applied as feasible solutions to meet the requirement of nonlinearity traffic data.

However, these approaches are still limited in capturing complex characteristics of traffic flows. In recent years, deep learning methods are prevalent and have achieved significant performance in traffic forecasting. For example, recurrent neural networks (RNNs) and long short-term memory networks (LSTMs) were studied to predict traffic sequences. Ma et al. [25] proposed to use LSTM network to capture nonlinear traffic dynamics, which shows superior capability for time series prediction with long temporal dependency. Similarly, Belhadi et al. [26] used RNNs to predict the long-term flows from multiple data sources. Ma et al. [27] established an improved LSTM model based on LSTM and bidirectional LSTM networks to overcome the large prediction errors. Such RNN-based methods take traffic flows from different locations as independent sequences while lacking the consideration of spatial dependencies. To further explore the spatial information, researchers divided road networks into several blocks and introduced the convolutional neural network (CNN) to extract spatial features [28]. Furthermore, due to the fact that graph architecture is conceivably more suited for modeling complex spatial relations, many existing works incorporate GNNs into deep learning-based traffic forecasting models to capture intrinsic spatial dependencies. Zhao et al. [7] proposed the temporal graph convolutional network (T-GCN), which is in combination with the GCN and gated recurrent unit (GRU). Zhu et al. [29] further model the external factors as dynamic and static attributes to enhance the performance of the spatiotemporal prediction models. Cui et al. [30] consider the transition of traffic states at consecutive time steps as a graph Markov process. In this way, the spatial-temporal relationships among the roadway links can be incorporated. Peng et al. [31] presented a novel spatial-temporal incidence dynamic graph neural networks framework, which combines the short-term, medium-term, and long-term historical traffic data to learn the temporal dependency comprehensively.

More recently, researchers further proposed to incorporate attention mechanism [32] to model complicated spatial-temporal correlations. Authors in [10, 33, 34] integrated the graph attention mechanism and gated temporal convolution to facilitate inductive spatial-temporal prediction problems. Tian et al. [11] employed a self-attention network on graph-structured spatial-temporal data to extract dynamic spatial dependencies. Here, the attention weight assignment principle is based on the intrinsic network information and traffic condition. Similarly, Shi et al. [35] used an encoder attention mechanism to model both the spatial and periodical dependencies. Li et al. [36] proposed an adaptive graph co-attention network (AGCAN), which consists of long-term and short-term graph attention modules to derive periodic patterns and respond to sudden traffic changes, respectively. Buroni et al. [37] built a multi-task learning model that introduces the multi-head attention mechanism to learn related tasks while improving generalization performance.

Generally, the above-mentioned methods mainly follow a uniform paradigm that leverages GNNs and RNN/LSTM-based structures to extract spatial and temporal features, respectively. Few of them have considered the connections and correlations between spatial and temporal dependencies. Moreover, due to the heterogeneity between temporal and spatial features, we cannot simply merge them in parallel and it is not applicable to adopt the same feature transformation function for spatial-temporal information as their contents vary from each other. In this paper, we are devoted to building a bridge between spatial and temporal dependencies and thus improving the accuracy of traffic forecasting.

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

TABLE 1 Summary of the main notations

Notation	Description
$\mathcal{G}$	the traffic road network
$\mathcal{V}$	the set of nodes
$\mathcal{E}$	the set of edges
N	the number of nodes
d	the number of node attributes
T	the historical time steps
T'	the future prediction time steps
$X_{\mathcal{G}}^t$	the observed graph signal at time step t
L	the length of series
k	the length of window
r	the tolerance threshold
H	the number of heads
A_S	the spatial adjacency matrix
A_T	the temporal correlation matrix
A_C	the spatial-temporal connection graph
A_{STC}	the spatial-temporal correlation graph

3 | PRELIMINARIES

In this section, we introduce some necessary notations and definitions that will be used in this paper. Then, we provide some related knowledge of cross-approximation entropy theory. Table 1 summarizes the main notations.

3.1 | Notations and problem formulation

We represent the road network topological structure as a weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A_S)$, where $\mathcal{V} = \{v_1, v_2, \cdots, v_N\}$ denotes the set of nodes, corresponding to N traffic sensors or roads. $\mathcal{E}$ denotes the set of edges and the spatial weighted adjacency matrix $A_S \in \mathbb{R}^{N \times N}$ contains the whole connectivity information (e.g. the node proximity measured by the actual road network distances).

In this paper, we denote the observed graph signal at time step t as $X_{\mathcal{G}}^t \in \mathbb{R}^{N \times d}$, where d is the number of node attributes (e.g. speed, volume etc.). The objective of this traffic forecasting task is to learn a mapping function f from previous T time period observations to predict future T' traffic conditions, which can be formulated as

$$\left[X_{\mathcal{G}}^{t+1}, X_{\mathcal{G}}^{t+2}, \dots, X_{\mathcal{G}}^{t+T'}\right] = f\left(\mathcal{G}; \left[X_{\mathcal{G}}^{t-T+1}, \dots, X_{\mathcal{G}}^{t}\right]\right). \quad (1)$$

3.2 | Cross-approximation entropy

The approximation entropy is defined to measure the complexity of a time sequence [18], which has been widely applied to physiological time-series analysis [38]. Later, Pincus et al. [39] further proposed the cross-approximation entropy (Cross-ApEn) algorithm to calculate the similarity of different time series patterns. Normally, large values of Cross-ApEn imply substantial differences or irregularities between two series.

Given two time series with equal length L, $x = [x_1, x_2, \dots, x_L]$ and $y = [y_1, y_2, \dots, y_L]$, the procedure to compute Cross-ApEn is as follows:

Firstly, series x and y are normalized into x^* and y^*, respectively

$$x_i^* = \frac{x_i - mean(x)}{SD(x)}, y_i^* = \frac{y_i - mean(y)}{SD(y)}, \quad (2)$$

where $mean(\cdot)$ and $SD(\cdot)$ represent the mean and standard deviation operations, respectively. Then, specify two windows with length k and construct two subvectors $U_i = [x_i^*, \dots, x_{i+k-1}^*]$ and $V_j = [y_j^*, \dots, y_{j+k-1}^*]$. Here, the distance between two vectors U_i and V_j is defined as $dist(U_i, V_j) = \|U_i - V_j\|_\infty$, where $\|\cdot\|_\infty$ denotes the infinite norm.

Subsequently, given a tolerance threshold r [18, 40], which conventionally takes the value of one-fifth times the covariance between two sequences, namely $0.2Cov(U_i, V_j)$, the ratio $C_i^k(r)$ is defined as

$$C_i^k(r) = \frac{T_i^k(r)}{L - k + 1}, \quad (3)$$

where $T_i^k(r)$ is the count of $j = 1, 2, \dots, L - k + 1$ that satisfy $dist(U_i, V_j) \le r$. Now, define

$$\psi^k(r)(y|x) = \frac{1}{L - k + 1} \sum_{i=1}^{L-k+1} \ln C_i^k(r), \quad (4)$$

and Cross-ApEn$(k, r, L)(y|x) = \psi^k(r)(y|x) - \psi^{k+1}(r)(y|x)$, $k \ge 1$.

In general, Cross-ApEn$(k, r, L)(y|x)$ measures the regularity or similarity between sequences x and y, which can be an intuition for extracting the temporal dependencies in traffic flow data. The corresponding pseudocode is shown in Algorithm 1.

4 | METHODOLOGY

In this section, we elaborate the proposed STCGCN architecture. Specifically, STCGCN is comprised of the input part, the spatial-temporal correlation graph convolutional layers, and the fully-connected layers for output. Each spatial-temporal correlation graph convolutional layer consists of multiple spatial-temporal correlation graph convolutional modules and a gated temporal convolution module in parallel.

4.1 | Temporal correlation extraction

As the temporal dependencies of traffic flows are complex and vary constantly, which presents various dynamic characteristics,

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

ALGORITHM 1 Cross-Approximation Entropy

```
Input: Two series $x = [x_1, x_2, \dots, x_L], y = [y_1, y_2, \dots, y_L]$
Output: Cross-ApEn$(k, r, L)(y|x)$
Specify window width $k$ and tolerance threshold $r$
Normalize $x$ and $y$ to $x_i^*$ and $y_i^*$, respectively
Construct $U_i$ and $V_j$ with $k$-dimension
for $i$ in $L - k + 1$ do
    for $j$ in $L - k + 1$ do
        $dist(U_i, V_j) = \|U_i - V_j\|_\infty$
        if $dist(U_i, V_j) \leq r$ then
            $T_i^k(r) \leftarrow increment$
        end if
        $C_i^k(r)$ and $C_i^{k+1}(r) \leftarrow compute$
    end for
end for
$\psi^k(r)(y|x)$ and $\psi^{k+1}(r)(y|x) \leftarrow compute$
Cross-ApEn$(k, r, L)(y|x) \leftarrow compute$
```

it is necessary to develop effective solutions to handle this. Traditional time series analysis methods including moving average [41] and exponential smoothing [42] are based on timesteps, which can hardly extract the complicated intrinsic characteristics and are sensitive to missing values. Other approaches such as Dynamic Time Warping (DTW) [43] overcomes the measurement problem of non-equal long sequences while taking more computing time and can only be implemented in time-insensitive scenarios.

Inspired by the approximate entropy [18, 40], which is among the most exploited nonlinear techniques to quantify the complexity and regularity of time series, we proposed to utilize cross-approximation entropy to measure the similarity and change patterns between two time series. According to the introduction of Cross-ApEn in Section 3.2, the calculation of Cross-ApEn requires only a small amount of data, which makes it possible to capture the dynamics of the cross-approximate entropy of long sequences. However, there exist a lot of redundant steps in these calculation procedures, which are inefficient and can cause high computational complexity. Therefore, we introduce the concept of binary distance matrix to further optimize this algorithm. Here, we retain some necessary notations and illustrations from Section 3.2 for better understanding.

For two sequences of length L, their binary distance matrix $D \in \mathbb{R}^{L \times L}$ is defined as

$$d_{ij} = \begin{cases} 1\,, & \left|x_i^* - y_j^*\right| < r \\ 0\,, & \left|x_i^* - y_j^*\right| \geq r \end{cases} \qquad i, j = 1 \sim L, \tag{5}$$

where d_{ij} represents the element in the i-th row and j-th column of D. Then, we can obtain any $C_i^k(r)$ through the following

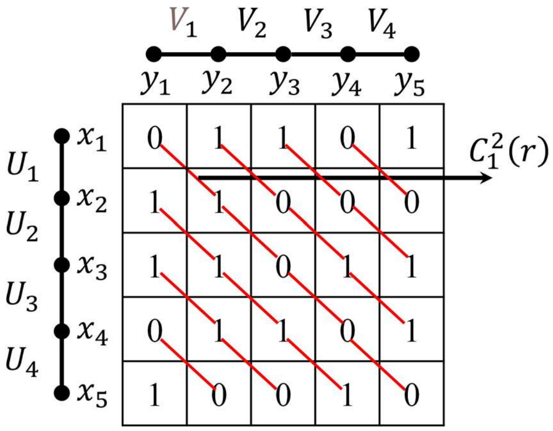


FIGURE 2 An example of binary distance matrix for sequences of length $L = 5$ and window length $k = 2$. The red backlashes between each diagonal element denote the logical conjunction

TABLE 2 Execution time of the optimized Cross-ApEn in comparison with the original algorithm

Length	Cross-ApEn [18]	Optimized Cross-ApEn (Ours)
L=10	0.01 s	<1e-4 s
L=100	0.21 s	0.01 s
L=1000	22.02 s	1.54 s
L=10000	2113.34 s	154.26 s

equation

$$C_i^k(r) = \sum_{j=1}^{L-k+1} d_{ij} \cap \dots \cap d_{(i-k+1)(j-k+1)}, \tag{6}$$

where $\cap$ denotes the logical conjunction. As shown in Figure 2, for sequences of length $L = 5$, we can easily obtain the value of $C_1^2(r)$ by summing all logical conjunction results (denoted as red backlashes) in the first row. Another example, determining whether the formula $dist(U_1, V_4) \leq r$ succeeds is equivalent to judging these two formulas $|x_1 - y_4| \leq r$ and $|x_2 - y_5| \leq r$ are valid at the same time, namely to determine whether $d_{14} \cap d_{25} = 1$ holds. Consequently, the calculation process of cross-approximation entropy is greatly simplified by introducing the distance matrix structure. The comparison of execution speed between the original and our optimized Cross-ApEn algorithm is given in Table 2. It can be observed that the proposed optimization strategy achieves orders of magnitude faster than the traditional iteration method, which greatly improves the computing efficiency for both short and long sequences.

4.2 | Spatial-temporal correlation graph construction

The purpose of generating temporal correlation graph is to build a more informative graph structure and to represent

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

ALGORITHM 2 Temporal Correlation Graph Generation

```
Input: $N$ time series from vertex set $\mathcal{V}(|\mathcal{V}| = N)$
Output: Temporal correlation Graph $A_T \in \mathbb{R}^{N \times N}$
$A_T \leftarrow$ initialize
for $i = 1, 2, \dots, N$ do
    for $j = 1, 2, \dots, N$ do
        $A_T(i, j) =$ Cross-ApEn$(v_i, v_j)$ (Alg. 1)
    end for
    Sort the smallest $m$ elements and their indexes
    $\mathbf{j} = \{j_1, j_2, \dots, j_m\}$ s.t.
    $A_T(i, j_1) \leq A_T(i, j_2) \leq \dots \leq A_T(i, j_m)$
    if $j' \in \mathbf{j}$ then
        $A_T(i, j') = A_T(j', i) = 1$
    else
        $A_T(i, j') = A_T(j', i) = 0$
    end if
end for
```

the spatial and temporal dependency with more genuine relationships than spatial graph alone. The detailed procedure for constructing a temporal correlation graph is given in Algorithm 2. Then, we proposed to incorporate the temporal graph into a novel spatial-temporal correlation graph, which can simplify the deep-learning architecture due to its comprehensiveness. Concretely, this fused graph already contains spatial information of each node with its neighbors and involves temporal evolution patterns for previous and future states.

However, the main obstacle of spatial-temporal correlation graph construction is how to establish a connection between temporal and spatial information. In fact, these two feature spaces are not completely irrelevant. Motivated by [32, 44], we leverage the multi-head attention mechanism to learn a weight matrix, which denotes the correlations between the temporal and spatial dependency. Specifically, as depicted in Figure 3, the input of scaled dot-product attention consists of queries Q, keys K, and values V with the dimension in d_q, d_k and d_v respectively. In this work, we employ the temporal correlation graph $A_T \in \mathbb{R}^{N \times N}$ as the query matrix and spatial adjacency matrix $A_S \in \mathbb{R}^{N \times N}$ (given by dataset) as the key matrix and the value matrix. We then compute the element-wise dot product of the query with all keys, scale the computation results by $\sqrt{d_k}$ and deploy a function to obtain the weights on the values. The formula is shown as follows:

$$Y = Attention(Q, K, V) = \sigma\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{7}$$

where σ denotes the softmax function and element $y_{i,j} \in Y$ denotes the learned spatial-temporal correlated representation of place in i-th row and j-th column. Afterward, in order to extract different representations at multiple perspectives, we further project the queries Q, keys K, and values V several times in parallel with three learnable parametric projection matrices W_Q^h, W_K^h, and W_V^h:

$$\widetilde{Q} = Q \cdot W_Q^h,\ \widetilde{K} = K \cdot W_K^h,\ \widetilde{V} = V \cdot W_V^h, \tag{8}$$

where h indicates the numerical order of heads. Besides, in order to capture spatial semantics from both local and global perspectives, we develop an attentive aggregation operation based on message propagation. Specifically, we define the aggregation of the features over the spatial topological structure with the following manipulation:

$$z_{(i,j) \leftarrow (i',j')} = \Big\|_{h=1}^{H} \left(\omega_{(i,j);(i',j')}^{h}\right) \cdot \widetilde{Y} \cdot W^p, \tag{9}$$

where $z_{(i,j) \leftarrow (i',j')}$ denotes that the features propagate from place (i', j') to (i, j) and $\widetilde{Y} = \sigma(\frac{\widetilde{Q}\widetilde{K}^T}{\sqrt{d_k}})\widetilde{V}$. Empirically, we deploy eight parallel attention layers ($H = 8$ as in [9, 32]) to derive spatial dependencies from different representation subspaces concurrently. Furthermore, W^p is the parameterized mapping matrix. The latent attentive relevance $\omega_{(i,j);(i',j')}^{h}$ can be described by:

$$\omega_{(i,j);(i',j')}^{h} = \frac{\exp\left(LR\left(\theta^T [\widetilde{y}_{i,j} \oplus \widetilde{y}_{i',j'}]\right)\right)}{\sum_{(i',j') \in \mathcal{N}(i,j)} \exp\left(LR\left(\theta^T \left[\widetilde{y}_{i,j} \oplus \widetilde{y}_{i',j'}\right]\right)\right)}, \tag{10}$$

where $\oplus$ denotes the concatenation between $\widetilde{y}_{i,j}$ and $\widetilde{y}_{i',j'}$ $(\widetilde{y}_{i,j} = y_{i,j} \cdot W^p)$. θ is the coefficient vector. $LR(\cdot)$ denotes the *LeakyReLU* function. Finally, the aggregation of spatial-temporal correlated feature embedding of each place $Z_{i,j}$ is defined as follows:

$$Z_{i,j} = \mathcal{F}\left(\sum_{(i',j') \in \mathcal{N}(i,j)} z_{(i,j) \leftarrow (i',j')}\right). \tag{11}$$

Subsequently, we design a connection graph $A_C \in \mathbb{R}^{N \times N}$ for associating spatial and temporal dependencies. Note that the element in i-th row and j-th column of A_C is $Z_{i,j}$. Figure 3 shows an example of spatial-temporal correlation graph A_{STC}. It consists of three types of $N \times N$ matrix, namely the temporal correlation graph A_T generated by Algorithm 2, the spatial adjacency graph A_S given by datasets, and the spatial-temporal connection graph A_C that builds a bridge between the spatial and temporal features. In this paper, we construct the spatial-temporal correlation graph $A_{STC} \in \mathbb{R}^{3N \times 3N}$ by concatenating three types of matrix together, which not only contain the topological information of complex traffic networks but also reflect the temporal dependencies between different spatial regions.

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

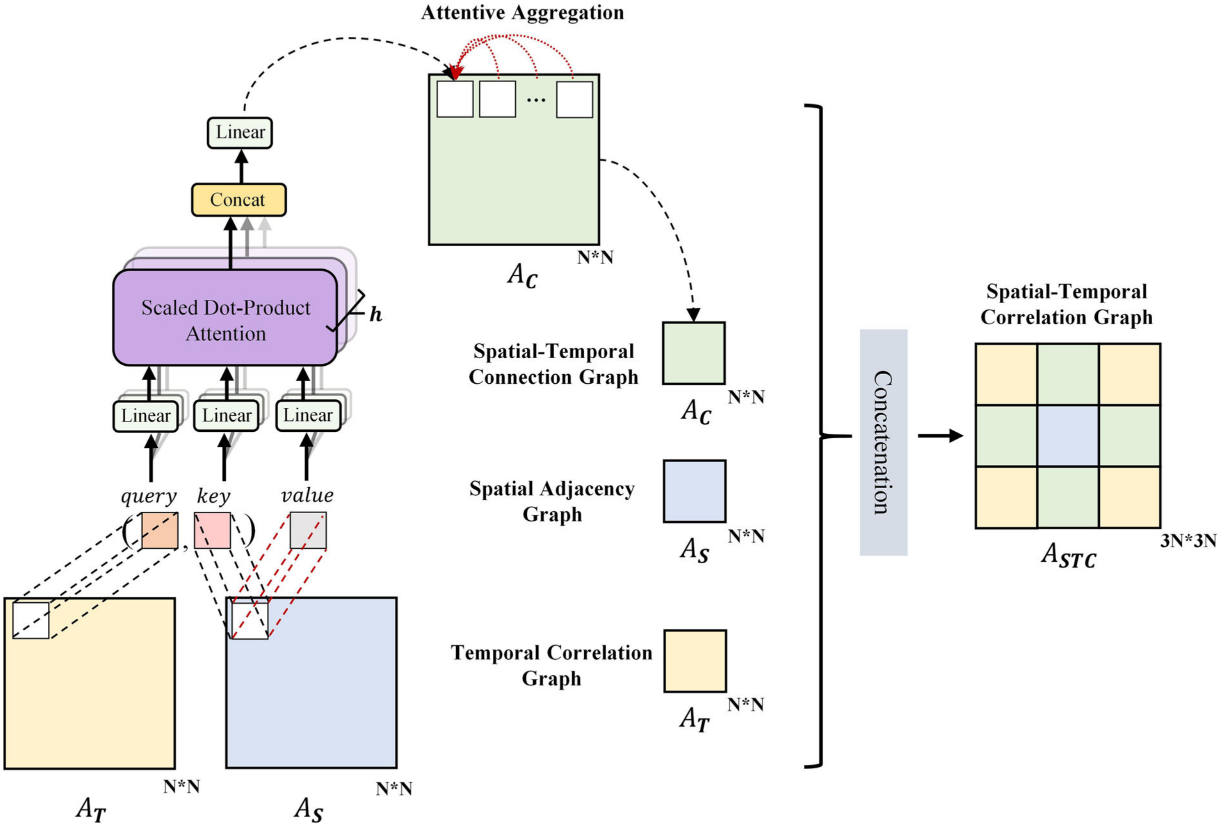


FIGURE 3 Construction of spatial-temporal correlation graph

4.3 | Spatial-temporal correlation graph convolutional module

We build a spatial-temporal correlation graph convolutional module (STCGCM) to further extract hidden spatial-temporal dependencies. The STCGCM consists of several graph convolutional operations, which enable each node to aggregate spatial dependency from A_S, temporal pattern similarity from A_T, and its spatial-temporal correlation from A_C. The input of the graph convolutional operation is the graph signal matrix of the spatial-temporal correlation graph. Different from the classical GCN that needs to calculate the graph laplacian, we define the graph convolutional operation in the vertex domain [14], which greatly simplifies and reduces the computation procedure. Besides, we leverage the gating mechanism to propagate features and the gated liner unit (GLU) is selected as the activation function. Graph convolutional operation is formulated as follows:

$$h^{(l+1)} = (\widetilde{A}h^{(l)}W_1 + b_1) \odot \sigma\left(\widetilde{A}h^{(l)}W_2 + b_2\right), \qquad (12)$$

where $h^{(l)}$ is the hidden state of l-th layer. $\widetilde{A}$ is the abbreviation of spatial-temporal correlation graph $A_{STC} \in \mathbb{R}^{3N\times 3N}$. $W_1, W_2 \in \mathbb{R}^{C\times C}$, $b_1, b_2 \in \mathbb{R}^{C}$ are all learnable parameters. $\sigma(\cdot)$ and $\odot$ are the softmax activation function and element-wise product, respectively. C denotes the dimension for both the input and output feature channels. The input of the first layer in each STCGCM $h^{(0)}$ is represented as:

$$h^{(0)} = \left[X_{\mathcal{G}}^{t}, \dots, X_{\mathcal{G}}^{t+3}\right] \in \mathbb{R}^{3\times N\times d\times C}, \qquad (13)$$

which is regarded as a high-dimension graph signal window sliced iteratively along the timeline. Moreover, we stack multiple graph convolutional operations to expand the aggregation area to capture more complicated spatial dependencies. In addition, the residual connections [45] are introduced to each graph convolutional operation to alleviate the gradient explosion problem. We then deploy max-pooling on the concatenation of each hidden state to further refine the features

$$h_M = MaxPool\left(\left[h^{(1)}, h^{(2)}, \dots, h^{(P)}\right]\right), \qquad (14)$$

where P is the number of layers. Finally, we further crop the output of the pooling layer, while retaining the features at the middle time step. This is because the graph convolutional operations have already aggregated the complicated information. A_S in the center of the diagonal provides information from neighbors. A_T in the corner provides the temporal pattern information of nodes. A_C located in horizontal and vertical positions gives a weighted matrix to connect the spatial and temporal information.

To sum up, the output of STCGCM is compacted into $h^{final} \in \mathbb{R}^{1\times N\times d\times C}$, which avoids existing too much redundant

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

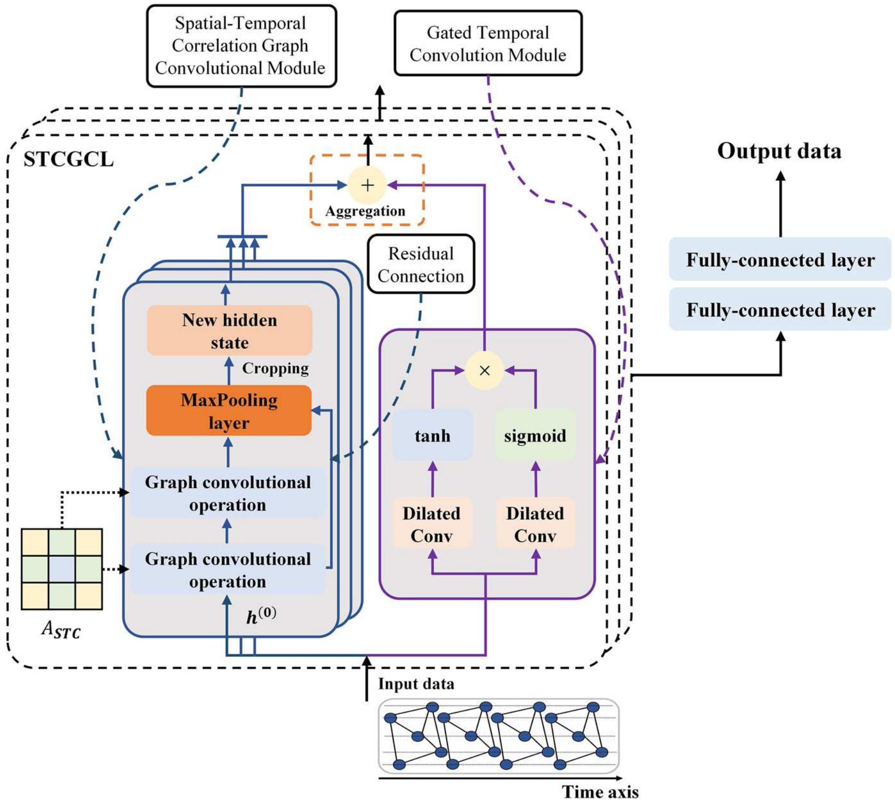


FIGURE 4 The overall structure of spatial-temporal correlation graph convolutional network

information. Furthermore, Figure 4 shows that the input data would be handled by several STCGCM in parallel, which allows the model to capture more complicated correlations.

4.4 | Gated temporal convolution module

The representation of A_{STC} enables the model to capture spatial-temporal correlations from a global perspective, while the long-range spatial-temporal dependencies of each node are also essential for obtaining accurate predictions. To address this, we introduce the gated temporal convolution and perform dilated convolution with a proper dilation rate to enlarge the receptive field along the timeline (see Figure 4). Compared with previous work like RNN-based approaches [46, 47], the gated convolution unit has a lighter structure and takes less time to calculate. Given the whole input data $X \in \mathbb{R}^{T\times N\times d\times C}$, the gated temporal convolution operation can be represented as follows:

$$Y_{output} = \psi(\Phi_1 * X + b_1) \odot \sigma(\Phi_2 * X + b_2), \tag{15}$$

where Φ_1 and Φ_2 are two 1D dilated convolution operations. ψ and σ denote the tanh and sigmoid functions, respectively. The notation of $\odot$ denotes Hadamard product. Through this way, certain messages can be determined whether it needs to be passed to the next layer, which shows advantages in handling sequence data.

4.5 | Other components

As shown in Figure 4, we integrate the stacked spatial-temporal correlation graph convolutional module and the gated temporal convolution module into a unified layer called the spatial-temporal correlation graph convolutional layer (STCGCL). The concatenation of each STCGCM output would be added with the gated temporal convolution output and sent to the next STCGCL. Note that the size of each STCGCL output is $\mathbb{R}^{(T-2)\times N\times d\times C}$. In the other words, each STCGCL would crop the input from T to $T-2$ in time dimensions, which also indicates that STCGCL could stack at most $\lfloor \frac{T}{2} \rfloor - 1$ layers. By stacking multiple STCGCLs, the spatial-temporal correlations and heterogeneity can be further exploited. Finally, we utilize two fully-connected layers to transform the output of the last STCGCL to possible predictions.

In this work, we select Huber loss [14, 48] as the loss function, which is a parameterized loss function for regression problems and has advantages in handling outliers compared to squared error loss. The formula is defined as follows:

$$L_\delta(Y,\hat{Y}) = \begin{cases} \frac{1}{2}(Y-\hat{Y})^2, & |Y-\hat{Y}| \le \delta \\ \delta \cdot |Y-\hat{Y}| - \frac{1}{2}\delta^2, & |Y-\hat{Y}| > \delta \end{cases}, \tag{16}$$

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

TABLE 3 Statistics of traffic flow prediction datasets

Datasets	#Nodes	#Edges	#Time steps	Time range
PEMS03	358	547	26208	9/1/2018 - 11/30/2018
PEMS04	307	340	16992	1/1/2018 - 2/28/2018
PEMS07	883	866	28224	5/1/2017 - 8/31/2017
PEMS08	170	295	17856	7/1/2016 - 8/31/2016
PEMS-BAY	325	2369	52116	1/1/2017 - 5/31/2017

where Y and $\hat{Y}$ are the ground truth and the predicted value, respectively. δ is a hyperparameter that governs the sensitivity of squared error loss.

5 | EXPERIMENTS

In this section, we provide comprehensive evaluations of the proposed STCGCN method and baselines in traffic forecasting tasks on five benchmark datasets. Due to the sparsity of the traffic flow data, we aggregate them into 5-min intervals, which means there are 12 points for one hour. Here, all input data are standardized by Z-score normalization for better performance. All experiments are conducted five times independently.

5.1 | Datasets and experimental settings

Five public traffic network datasets, including PEMS03, PEMS04, PEMS07, PEMS08, and PEMS-BAY, are selected for performance evaluation [14, 49]. We divide the datasets with ratio 6:2:2 into training sets, validation sets, and testing sets. The detailed information of the datasets is presented in Table 3. One hour 12 continuous time steps historical data is used to predict the traffic conditions of the next 12 time steps. Figure 5 shows the geographical location of sensors in dataset PEMS-BAY.

To make a fair comparison, all experiments are conducted using the Windows (64-bit) PC with Intel Core i5-9300HF CPU 2.4GHz, 16GB RAM, and NVIDIA GeForce GTX 1660Ti 6G GPU. The experimental programming language is Python 3.7. Mxnet is used as an experimental software framework. Considering the size of spatial-temporal correlation graph, we deploy five STCGCLs in this model, where each contains nine independent spatial-temporal correlation graph convolutional modules and one gated temporal convolution module with dilated rate 2. The sensitivity threshold parameter of loss function δ is 1. The filter size in all convolution operations is 64. We train the model using the Adam optimization algorithm with a learning rate of 0.001 and batch size of 32. The training epoch is set as 200. As for the comparison algorithms, we use the original implementations and parameter settings released by the authors.

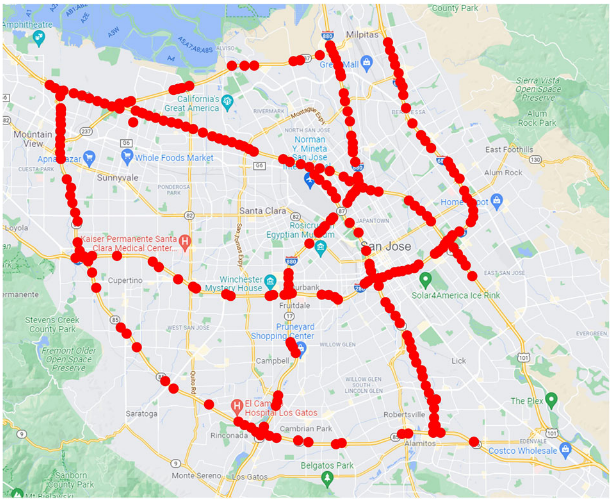


FIGURE 5 The geographical distribution of sensors in the PEMS-BAY dataset

5.2 | Baseline description

To demonstrate the effectiveness of our proposed method, we make comparisons with the following approaches:

- SVR [24]: Support vector regression. A machine learning method for regression problems.
- FC-LSTM [50]: Long short-term memory network, a sequence-to-sequence model based on the recurrent neural network with fully-connected LSTM hidden units.
- DCRNN [49]: Diffusion convolutional recurrent neural network, which uses diffusion graph convolutional networks and the encoder-decoder structure to capture the spatial and temporal dependencies, respectively.
- STGCN [6]: Spatio-temporal graph convolution networks, which integrate the gated temporal convolution unit into graph convolution blocks.
- GraphWaveNet [17]: Graph WaveNet combines an adaptive adjacency matrix into graph convolution and utilizes stacked 1D convolution units to capture the spatial-temporal dependency.
- STSGCN [14]: Spatial-temporal synchronous graph convolutional networks, which effectively capture the localized and long-range spatial-temporal dependencies through a spatial-temporal synchronous modeling mechanism.
- GMAN [9]: Graph multi-attention network, an encoder-decoder architecture with multiple spatial-temporal attention blocks. The transform attention layers enable modeling the impact of the spatio-temporal factors on complex traffic conditions.
- ST-MGAT [51]: Spatial-temporal multi-head graph attention network, which consists of temporal convolution blocks and graph attention networks for capturing the dynamic temporal correlations and spatial relations between nodes, respectively.

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

5.3 | Evaluation metrics

In the experiment, three commonly used regression evaluation metrics are adopted to evaluate the performance of each method.

- Mean absolute error (MAE):

$$MAE = \frac{1}{MN}\sum_{t=1}^{M}\sum_{i=1}^{N}\left|y_i^t - \hat{y}_i^t\right|. \tag{17}$$

- Mean absolute percentage error (MAPE):

$$MAPE = \frac{1}{MN}\sum_{t=1}^{M}\sum_{i=1}^{N}\left|\frac{y_i^t - \hat{y}_i^t}{y_i^t}\right|. \tag{18}$$

- Root mean squared error (RMSE):

$$RMSE = \sqrt{\frac{1}{MN}\sum_{t=1}^{M}\sum_{i=1}^{N}(y_i^t - \hat{y}_i^t)^2}, \tag{19}$$

where y_i^t and $\hat{y}_i^t$ are the observed and predicted traffic data at time t on sensor i, respectively. M is the number of temporal samples. N is the number of all sensors on the road.

5.4 | Experimental results and analysis

To verify the effectiveness of the STCGCN model in the traffic forecasting task, we conduct four types of experiments on PEMS03, PEMS04, PEMS07, PEMS08, and PEMS-BAY datasets.

(i) Comparison with baseline methods: We evaluate the performance of STCGCN with eight baseline methods on five real-world traffic network datasets. Three indicators are utilized to measure the performance of the model. The traffic flow information in the next 15, 30, and 60 min is predicted.
(ii) Ablation experiments: To quantitatively verify the effectiveness of each component, we design three variants of the STCGCN model with different configurations and test their performance on PEMS08 and PEMS-BAY datasets for one hour traffic prediction.
(iii) Impact of hyperparameters: We investigate four hyperparameters, that is, the number of filters, the batch size, the number of training epochs, and the number of historical time intervals, to find the best experimental setting of the model.
(iv) Perturbation analysis: We introduce the Gaussian noise and random noise that obeys Poisson distribution to evaluate the robustness of the proposed STCGCN. The PEMS08 and PEMS-BAY datasets are selected as the test benches.

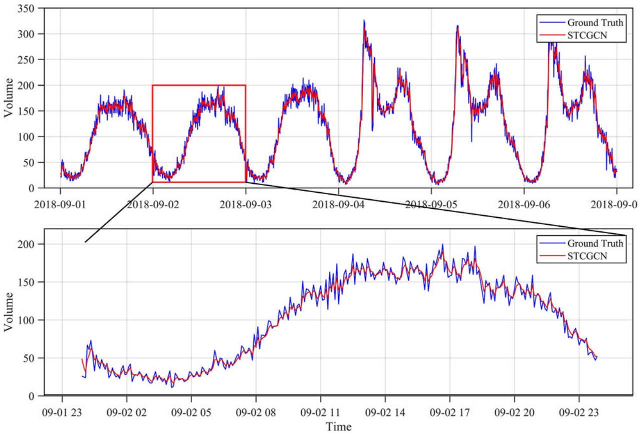


FIGURE 6 The visualization results for the 15 min prediction horizon

5.4.1 | Comparison with baseline methods

The experimental results for different prediction horizons (15 min, 30 min, and 60 min) of the comparison with baseline methods are reported in Table 4. Concretely, for the prediction horizon of 15 min, the RMSE of STCGCN is approximately lower than the suboptimal methods by 0.95%, 5.43%, 3.94%, 0.08%, and 1.84% on PEMS03, PEMS04, PEMS07, PEMS08, and PEMS-BAY, respectively. With respect to the 30-min time sequences, the RMSE values of GMAN and ST-MGAT are slightly lower than our model by margins of 1.28% and 1.10% on PEMS04 and PEMS-BAY, respectively. For the prediction horizon of 60 min, the RMSE of STCGCN is approximately 5.47% and 10.34% lower than those of GMAN and ST-MGAT on PEMS04, respectively, which verifies the effectiveness of the stacked STCGCLs in long-term spatial-temporal dependencies extraction. Early methods such as SVR and FC-LSTM perform poorly in three horizons of prediction and have a big gap with our model, which is mainly caused by that these methods only take temporal features into consideration while lacking the exploration of spatial information. Besides, typical spatiotemporal forecasting methods, including STGCN, Graph WaveNet, and STSGCN, achieve poorer results than our model by margins of 8.32%, 19.50%, and 9.02% on PEMS07 in terms of RMSE. Similar conclusions could be drawn for other datasets. These approaches integrate spatial and temporal information without consideration of their correlations, thereby leading to subpar performance. In general, the proposed STCGCN model can obtain the optimal prediction performance of all metrics in most scenarios, which proves the validity and superiority of STCGCN in spatial-temporal traffic flow prediction tasks.

To illustrate the predictive ability of the proposed STCGCN more intuitively, we also conduct visualization experiments, which compare and analyze the true traffic volume and the predicted results of the spatial-temporal correlation graph convolutional network. Based on the historical one hour data, the predicted results for the next 15 min, 30 min, and 60 min are shown in Figures 6, 7, and 8, respectively. The upper subfigure in each figure is the forecasting result on PEMS03 from 1 September 2018, to 6 September 2018. The lower subfigure exhibits the

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

TABLE 4 Performance comparison among baseline methods on PEMS03, PEMS04, PEMS07, PEMS08, and PEMS-BAY datasets. The best performing methods are highlighted in **bold**

		15 min			30 min			60 min		
Datasets	**Methods**	**MAE**	**MAPE(%)**	**RMSE**	**MAE**	**MAPE(%)**	**RMSE**	**MAE**	**MAPE(%)**	**RMSE**
PEMS03	SVR	16.63	16.77	26.87	19.40	18.18	32.87	21.97	21.51	35.29
	FC-LSTM	17.82	17.87	28.24	20.17	20.58	33.40	21.33	23.33	35.11
	DCRNN	15.35	16.11	26.12	18.31	17.73	31.29	18.18	18.91	30.31
	STGCN	15.13	15.34	25.61	17.64	17.11	30.68	17.49	17.15	30.12
	Graph WaveNet	13.82	13.03	23.54	16.20	15.66	28.29	19.85	19.31	32.94
	STSGCN	12.97	12.96	22.37	15.88	**14.48**	27.01	17.48	16.78	29.21
	GMAN	12.68	13.45	22.34	15.82	14.79	26.91	17.62	18.63	31.37
	ST-MGAT	13.35	13.10	22.39	16.22	15.33	27.13	18.89	19.12	32.20
	STCGCN (Ours)	**12.10**	**12.87**	**22.13**	**15.51**	14.56	**26.63**	**16.53**	**16.18**	**28.07**
PEMS04	SVR	21.24	14.39	28.23	26.83	17.31	34.66	28.70	19.20	44.56
	FC-LSTM	23.23	15.16	30.31	25.62	17.70	38.83	27.14	18.20	41.59
	DCRNN	21.15	13.21	26.10	22.71	15.32	34.41	24.70	17.12	38.12
	STGCN	21.28	13.27	25.54	22.43	15.03	32.91	22.70	14.59	35.55
	Graph WaveNet	19.71	12.55	25.12	22.15	14.68	31.32	25.45	17.29	39.70
	STSGCN	15.43	11.03	21.65	18.34	11.77	27.76	21.19	13.90	33.65
	GMAN	16.20	10.68	22.58	17.81	12.70	**27.41**	21.55	14.23	33.56
	ST-MGAT	15.77	10.82	22.80	16.21	11.55	28.84	22.60	15.24	35.11
	STCGCN (Ours)	**13.53**	**8.71**	**20.54**	**15.50**	**11.03**	27.76	**19.53**	**12.92**	**31.82**
PEMS07	SVR	24.73	11.34	32.77	26.23	13.45	46.32	32.49	14.26	50.22
	FC-LSTM	24.66	10.52	29.93	28.28	13.20	45.17	29.98	13.20	45.84
	DCRNN	21.03	8.92	32.83	23.94	10.28	37.05	25.30	11.66	38.58
	STGCN	22.59	8.96	33.81	24.00	10.30	37.14	25.38	11.08	38.78
	Graph WaveNet	19.76	8.09	30.89	22.15	8.94	34.53	26.85	12.12	42.78
	STSGCN	18.53	8.11	31.14	21.96	9.03	34.02	24.26	10.21	39.03
	GMAN	16.45	7.17	27.68	19.64	7.98	31.54	**22.04**	9.43	36.18
	ST-MGAT	17.33	7.58	27.41	19.91	8.15	32.28	23.89	11.23	37.38
	STCGCN (Ours)	**16.16**	**6.89**	**26.37**	**19.55**	**7.94**	**31.34**	22.07	**9.21**	**35.80**
PEMS08	SVR	18.18	12.23	29.31	21.45	14.01	32.40	23.25	14.64	36.16
	FC-LSTM	19.21	12.44	29.79	21.03	13.77	32.19	22.20	14.20	34.06
	DCRNN	12.35	8.63	23.27	14.71	10.05	25.35	17.86	11.45	27.83
	STGCN	11.17	8.31	22.98	14.30	10.23	25.33	18.02	11.40	27.82
	Graph WaveNet	10.89	7.92	20.61	14.12	10.31	23.88	19.13	12.68	31.05
	STSGCN	**9.81**	6.77	18.83	13.59	8.91	22.44	17.13	10.96	26.80
	GMAN	9.86	7.01	19.77	13.68	9.33	22.89	16.76	10.43	25.68
	ST-MGAT	10.01	7.95	20.13	14.34	9.88	23.45	16.27	11.12	25.56
	STCGCN (Ours)	9.83	**6.71**	**18.68**	**13.14**	**8.81**	**22.31**	**15.87**	**10.18**	**24.97**
PEMS-BAY	SVR	1.85	3.83	3.62	2.48	5.50	5.28	3.42	8.43	6.50
	FC-LSTM	2.11	4.80	4.19	2.20	5.23	4.55	2.46	5.77	5.02
	DCRNN	1.53	2.90	2.95	1.77	3.94	3.97	2.70	5.02	4.92
	STGCN	1.42	2.96	2.97	1.81	4.17	4.27	2.51	5.78	5.74
	Graph WaveNet	1.33	2.73	2.74	1.66	3.67	3.73	2.17	4.89	4.76
	STSGCN	1.30	2.88	2.81	1.73	3.75	3.77	2.09	4.92	4.74
	GMAN	1.34	2.82	2.81	1.62	3.65	3.73	1.93	4.68	4.52
	ST-MGAT	1.31	2.93	2.77	1.61	3.82	**3.67**	1.91	4.69	4.46
	STCGCN (Ours)	**1.26**	**2.67**	**2.72**	**1.52**	**3.63**	3.71	**1.89**	**4.66**	**4.43**

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

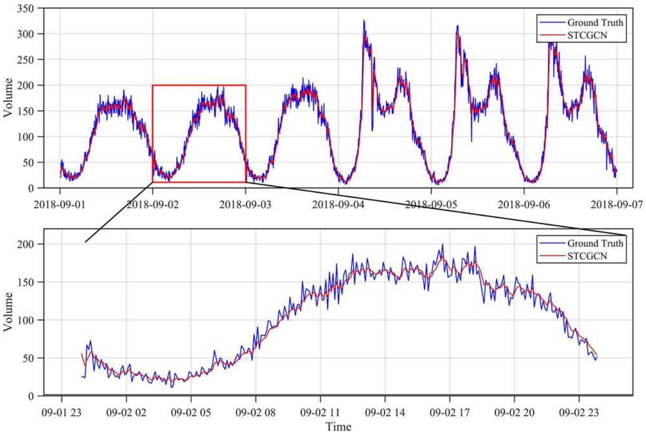


FIGURE 7 The visualization results for the 30 min prediction horizon

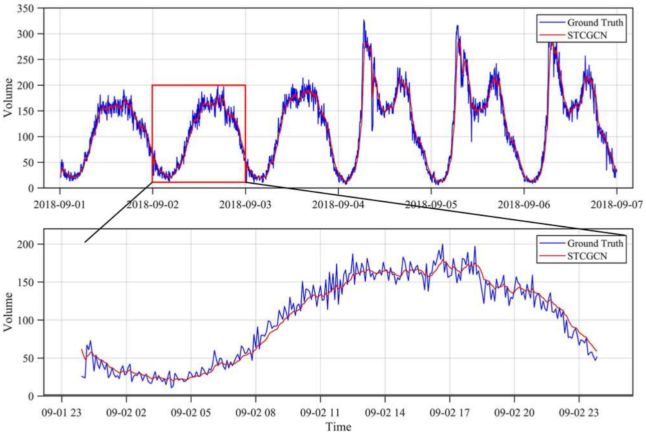


FIGURE 8 The visualization results for the 60 min prediction horizon

forecasting result for 2 September 2018. We can conclude that the proposed model can greatly handle not only short-term prediction but also long-term prediction. Meanwhile, the changing trend of the predicted results is highly consistent with the real observed data. Furthermore, it can be observed that the performance of 15 min prediction is better than 30 and 60 min, which indicates that the model is able to better capture short-term dependencies while losing some information in long-term dependency extraction. Due to the limitation of the dataset, STCGCN can hardly respond to spike data points, which may be caused by car accidents, weather, or other combinations of sudden incidents.

5.4.2 | Ablation experiments

There are three crucial designs in our proposed STCGCN, namely the multi-head attention-based feature aggregation module for spatial-temporal correlation extraction, the spatial-temporal correlation graph convolutional module for capturing hidden spatial-temporal dependencies, and the gated temporal convolution module for learning long-range spatial-temporal dependency. For brevity, we use "MHA", "GCN", and "GTC" to denote these modules, respectively. To probe into the effectiveness of different components of STCGCN, we design three variants with seven combinations and conduct ablation experiments on the PEMS08 and PEMS-BAY datasets. The detailed settings of these three variants are described as below:

TABLE 5 Ablation experiments on different configurations of modules

Datasets	Configuration	MAE	MAPE%	RMSE
PEMS08	STCGCN	**15.87**	**10.18**	**24.97**
	w/o [MHA]	16.78	10.67	26.23
	w/o [GCN]	18.23	11.77	28.43
	w/o [GTC]	19.91	12.33	29.52
	w/o [MHA, GCN]	19.65	12.15	29.01
	w/o [MHA, GTC]	20.16	12.87	31.03
	w/o [GCN, GTC]	21.53	13.13	32.34
	w/o [MHA, GCN, GTC]	23.41	14.79	34.68
PEMS-BAY	STCGCN	**1.89**	**4.66**	**4.43**
	w/o [MHA]	2.24	5.23	5.18
	w/o [GCN]	2.57	5.61	5.33
	w/o [GTC]	3.31	8.03	6.21
	w/o [MHA, GCN]	3.42	8.16	6.53
	w/o [MHA, GTC]	3.56	8.29	6.87
	w/o [GCN, GTC]	4.59	8.96	7.34
	w/o [MHA, GCN, GTC]	4.78	9.63	7.81

- STCGCN w/o MHA: In this variant, we remove the multi-head attention feature aggregation module in connecting the spatial and temporal information. Instead, we adopt an identity matrix with the same dimension to replace the spatial-temporal connection graph for constructing the spatial-temporal correlation graph.
- STCGCN w/o GCN: In this variant, we replace all graph convolutional operations in the spatial-temporal correlation graph convolutional module to fully-connected layers, which aims to test the effectiveness of graph convolutional operation in extracting complex spatial-temporal features.
- STCGCN w/o GTC: In this variant, we remove the gated temporal convolution module entirely to demonstrate the importance of temporal convolution in extracting long-range dependencies. Here, the input data will be directly sent into the spatial-temporal correlation graph convolutional module for global spatial-temporal dependency extraction.

Table 5 reports the results of the ablation experiments, in which "Configuration" represents seven different combinations of the three models. It can be observed that the model equipped with MHA surpasses the model without it, which shows the necessity of modeling the spatial-temporal correlations and heterogeneities in traffic prediction. Furthermore, the model equipped with GCN for each STCGCL outperforms that using the fully-connected layers by a large margin, which demonstrates the effectiveness of graph convolutional

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

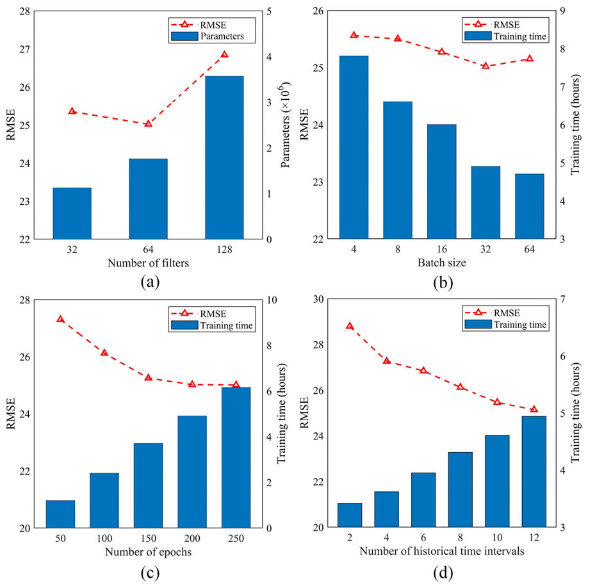


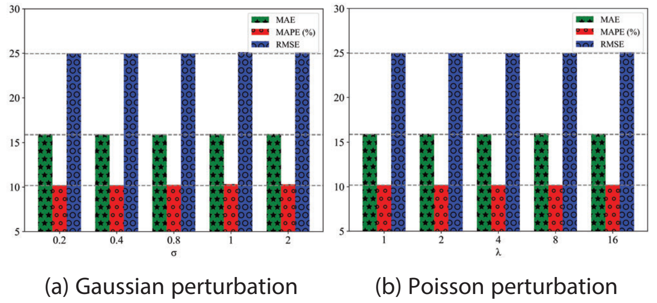
FIGURE 9 The impact of four hyperparameters in STCGCN. (a) RMSE with respect to the number of filters. (b) RMSE with the respect to the batch size. (c) RMSE with respect to the number of training epochs. (d) RMSE with respect to the number of historical time intervals

operation in extracting spatial-temporal dependencies. For the gated temporal convolution module, it could improve the long-range learning ability of STCGCN significantly. Those models equipped with GTC modules perform on average 16.09% and 30.81% lower than those without GTC in terms of RMSE on PEMS08 and PEMS-BAY, respectively. Obviously, the results show that all modules contribute to the improvement of model performance and enhance the capacity of handling spatial-temporal data.

5.4.3 | Impact of hyperparameters

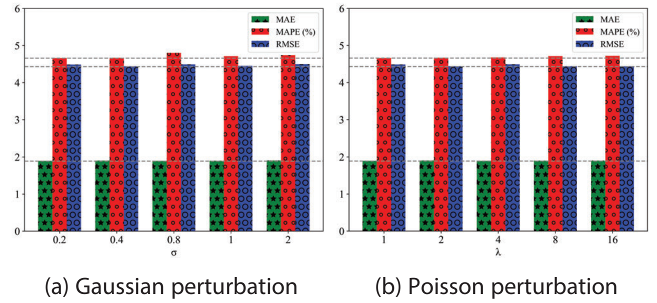
To find the best settings of the STCGCN model, we further analyze the impact of four hyperparameters including the number of filters, the batch size, the number of training epochs, and the number of historical time intervals. As shown in Figure 9, we choose RMSE as the evaluation indicator to measure the impact of hyperparameters and change one hyperparameter while remaining the other three hyperparameters unchanged.

Figure 9a presents the impact of the number of filters. The results are intuitive that adding a certain number of filters can improve the performance of the model, while too many filters will increase the number of parameters and burden the computational costs. As shown in Figure 9b, when setting the batch size as 32, the test performance reaches optimal. On the contrary, as the batch size increases further, the RMSE increase, which indicates that setting an appropriate batch size in the training stage can improve the model performance. Similarly shown in Figure 9c, the effect of the model improves as the training epoch increase, whereas longer training time is needed. However, excessive training epochs may cause an over-smoothing problem. We observe that when the training epochs reach 200, the RMSE almost stops decreasing. Figure 9d shows the impact of the historical time intervals. It can be observed that as the number of historical time intervals increases, the more temporal information is considered, the better predictive performance can be achieved.



(a) Gaussian perturbation (b) Poisson perturbation

FIGURE 10 PEMS08: Perturbation analysis. (a) Gaussian perturbation. (b) Poisson perturbation



(a) Gaussian perturbation (b) Poisson perturbation

FIGURE 11 PEMS-BAY: Perturbation analysis. (a) Gaussian perturbation. (b) Poisson perturbation

5.4.4 | Perturbation analysis

Noise is ubiquitous in real-world traffic forecasting applications. Therefore, perturbation analysis is conducted on datasets PEMS08 and PEMS-BAY for one hour prediction to verify the robustness of STCGCN. Specifically, we introduce two types of random noises to the traffic data, namely the random noise obeys Gaussian distribution $N(0, \sigma^2)$, where $\sigma \in [0.2, 0.4, 0.8, 1, 2]$ and Poisson distribution $P(\lambda)$, where $\lambda \in [1, 2, 4, 8, 16]$. Note that all the noise matrix values are normalized to [0,1].

The experimental results based on the PEMS08 dataset are shown in Figure 10. The results of adding Gaussian noise and Poisson noise are shown in Figures 10a and 10b, respectively, where the gray dashed lines in each figure represent the performance of STCGCN with no external noise added. It can be observed that the values of different evaluation metrics almost remain the same regardless of the changes in σ or λ, which indicates that the proposed STCGCN can well resist and handle noise with little performance fluctuation. Similarly, the experimental results shown in Figure 11 are consistent with the results in Figure 10. Overall, the changes in evaluation metrics across

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

different noise settings are negligible, which demonstrates the robustness of the STCGCN model.

6 | CONCLUSION

This paper addresses the problem that the existing urban traffic forecasting methods lack the construction of an informative graph that cannot comprehensively consider the spatial-temporal correlations and proposes a novel spatial-temporal correlation graph convolutional network (STCGCN). The STCGCN model is mainly comprised of three types of modules, namely, the multi-head attention-based feature aggregation module for spatial-temporal correlation graph construction, the spatial-temporal correlation graph convolutional module for extracting global spatial-temporal dependency, the gated temporal convolution module for long-range spatial-temporal dependency extraction. Such architecture enables learning spatial-temporal heterogeneity and modeling traffic patterns more accurately. Extensive experiments have been conducted on five real-world datasets. The results show that STCGCN outperforms the baselines in traffic forecasting tasks, thereby demonstrating its superiority in exploiting the correlations between spatial and temporal information. In future work, we will further apply our model to the general dynamic graphs and introduce graph theory-based techniques to excavate the operation mechanism of the traffic network.

AUTHOR CONTRIBUTIONS

Ru Huang: Conceptualization, funding acquisition, methodology, supervision, writing - review and editing. Zijian Chen: Conceptualization, data curation, formal analysis, methodology, resources, software, validation, visualization, writing - original draft. Guangtao Zhai: Formal analysis, writing - review and editing. Jianhua He: Formal analysis, writing - review and editing. Xiaoli Chu(GE): Formal analysis, writing - review and editing.

ACKNOWLEDGEMENTS

This research is supported in part by the National Natural Science Foundation of China under Grants 61673178 and 61922063; in part by the Natural Science Foundation of Shanghai under Grant 20ZR1413800; in part by European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie Grant Agreement Nos 824019 and 101022280.

CONFLICT OF INTEREST

The authors declare that they have no competing interests.

DATA AVAILABILITY STATEMENT

Some or all data, models, or code generated or used during the study are available from the author by request.

ORCID

Ru Huang https://orcid.org/0000-0001-7545-0987

REFERENCES

1. Hong, W.C.: Traffic flow forecasting by seasonal svr with chaotic simulated annealing algorithm. Neurocomputing 74(12-13), 2096–2107 (2011)
2. Oh, S., Byon, Y.J., Yeo, H.: Improvement of search strategy with k-nearest neighbors approach for traffic state prediction. IEEE Trans. Intell. Transp. Syst. 17(4), 1146–1156 (2015)
3. Tang, J., Liang, J., Liu, F., Hao, J., Wang, Y.: Multi-community passenger demand prediction at region level based on spatio-temporal graph convolutional network. Transport. Res. Part C: Emerg. Technol. 124, 102951 (2021)
4. Zhang, X., Huang, C., Xu, Y., Xia, L.: Spatial-temporal convolutional graph attention networks for citywide traffic flow forecasting. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 1853–1862. ACM, New York (2020)
5. Zhang, C., Zhang, S., James, J., Yu, S.: Fastgnn: A topological information protected federated learning approach for traffic speed forecasting. IEEE Trans. Ind. Inf. 17(12), 8464–8474 (2021)
6. Yu, B., Yin, H., Zhu, Z.: Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence, pp. 3634–3640. Springer, Cham (2018)
7. Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., Deng, M., Li, H.: T-gcn: A temporal graph convolutional network for traffic prediction. IEEE Trans. Intell. Transp. Syst. 21(9), 3848–3858 (2019)
8. Li, D., Lasenby, J.: Spatiotemporal attention-based graph convolution network for segment-level traffic prediction. IEEE Trans. Intell. Transp. Syst. 23(7), 8337–8345 (2022)
9. Zheng, C., Fan, X., Wang, C., Qi, J.: Gman: A graph multi-attention network for traffic prediction. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 1234–1241. AAAI Press, Menlo Park, CA (2020)
10. Wang, B., Wang, J.: St-mgat: Spatio-temporal multi-head graph attention network for traffic prediction. Phys. A 603, 127762 (2022)
11. Tian, C., Chan, W.K.: Spatial-temporal attention wavenet: A deep learning framework for traffic prediction considering spatial-temporal dependencies. IET Intel. Transport Syst. 15(4), 549–561 (2021)
12. Chai, D., Wang, L., Yang, Q.: Bike flow prediction with multi-graph convolutional networks. In: Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 397–400. ACM, New York (2018)
13. Chen, W., Chen, L., Xie, Y., Cao, W., Gao, Y., Feng, X.: Multi-range attentive bicomponent graph convolutional network for traffic forecasting. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 3529–3536. AAAI Press, Menlo Park, CA (2020)
14. Song, C., Lin, Y., Guo, S., Wan, H.: Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 914–921. AAAI Press, Menlo Park, CA (2020)
15. Feng, D., Wu, Z., Zhang, J., Wu, Z.: Dynamic global-local spatial-temporal network for traffic speed prediction. IEEE Access 8, 209296–209307 (2020)
16. Ren, Y., Zhao, D., Luo, D., Ma, H., Duan, P.: Global-local temporal convolutional network for traffic flow prediction. IEEE Trans. Intell. Transp. Syst. 23(2), 1578–1584 (2022)
17. Wu, Z., Pan, S., Long, G., Jiang, J., Zhang, C.: Graph wavenet for deep spatial-temporal graph modeling. In: The 28th International Joint Conference on Artificial Intelligence (IJCAI). Springer, Cham (2019)
18. Pincus, S.M.: Approximate entropy as a measure of system complexity. Proc. Natl. Acad. Sci. 88(6), 2297–2301 (1991)
19. Smith, B.L., Demetsky, M.J.: Traffic flow forecasting: comparison of modeling approaches. J. Transp. Eng. 123(4), 261–266 (1997)
20. Hamed, M.M., Al.Masaeid, H.R., Said, Z.M.B.: Short-term prediction of traffic volume in urban arterials. J. Transp. Eng. 121(3), 249–254 (1995)
21. Ahmed, M.S., Cook, A.R.: Analysis of freeway traffic time-series data by using Box-Jenkins techniques. Transport. Res. Rec. 722, 1–9 (1979)

17519578, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

22. Chandra, S.R., Al-Deek, H.: Predictions of freeway traffic speeds and volumes using vector autoregressive models. J. Intell. Transp. Syst. 13(2), 53–72 (2009)
23. Davis, G.A., Nihan, N.L.: Nonparametric regression and short-term freeway traffic forecasting. J. Transp. Eng. 117(2), 178–188 (1991)
24. Drucker, H., Burges, C.J., Kaufman, L., Smola, A., Vapnik, V., et al.: Support vector regression machines. In: Advances in Neural Information Processing Systems, vol. 9, pp. 155–161. MIT Press, Cambridge, MA (1997)
25. Ma, X., Tao, Z., Wang, Y., Yu, H., Wang, Y.: Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. Transport. Res. Part C: Emerg. Technolog. 54, 187–197 (2015)
26. Belhadi, A., Djenouri, Y., Djenouri, D., Lin, J.C.W.: A recurrent neural network for urban long-term traffic flow forecasting. Appl. Intell. 50(10), 3252–3265 (2020)
27. Ma, C., Dai, G., Zhou, J.: Short-term traffic flow prediction for urban road sections based on time series analysis and lstm_bilstm method. IEEE Trans. Intell. Transp. Syst. 23(6), 5615–5624 (2022)
28. Ma, X., Dai, Z., He, Z., Ma, J., Wang, Y., Wang, Y.: Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction. Sensors 17(4), 818 (2017)
29. Zhu, J., Wang, Q., Tao, C., Deng, H., Zhao, L., Li, H.: Ast-gcn: Attribute-augmented spatiotemporal graph convolutional network for traffic forecasting. IEEE Access 9, 35973–35983 (2021)
30. Cui, Z., Lin, L., Pu, Z., Wang, Y.: Graph markov network for traffic forecasting with missing data. Transport. Res. Part C: Emerg. Technolog. 117, 102671 (2020)
31. Peng, H., Wang, H., Du, B., Bhuiyan, M.Z.A., Ma, H., Liu, J., Wang, L., Yang, Z., Du, L., Wang, S., Yu, P.S.: Spatial temporal incidence dynamic graph neural networks for traffic flow forecasting. Inf. Sci. 521, 277–290 (2020)
32. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30. MIT Press, Cambridge, MA (2017)
33. Guo, G., Yuan, W.: Short-term traffic speed forecasting based on graph attention temporal convolutional networks. Neurocomputing 410, 387–393 (2020)
34. Kong, X., Xing, W., Wei, X., Bao, P., Zhang, J., Lu, W.: Stgat: spatial-temporal graph attention networks for traffic flow forecasting. IEEE Access 8, 134363–134372 (2020)
35. Shi, X., Qi, H., Shen, Y., Wu, G., Yin, B.: A spatial–temporal attention approach for traffic prediction. IEEE Trans. Intell. Transp. Syst. 22(8), 4909–4918 (2021)
36. Li, B., Guo, T., Wang, Y., Gandomi, A.H., Chen, F.: Adaptive graph co-attention networks for traffic forecasting. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 263–276. Springer, Cham (2021)
37. Buroni, G., Lebichot, B., Bontempi, G.: Ast-mtl: An attention-based multi-task learning strategy for traffic forecasting. IEEE Access 9, 77359–77370 (2021)
38. Pincus, S.M., Goldberger, A.L.: Physiological time-series analysis: what does regularity quantify? Am. J. Physiol.-Heart. Circulatory Physiol. 266(4), H1643–H1656 (1994)
39. Pincus, S., Singer, B.H.: Randomness and degrees of irregularity. Proc. Natl. Acad. Sci. 93(5), 2083–2088 (1996)
40. Pincus, S.: Approximate entropy (apen) as a complexity measure. Chaos: Interdisciplin. J. Nonlinear Sci. 5(1), 110–117 (1995)
41. Crato, N., Ray, B.K.: Model selection and forecasting for long-range dependent processes. J. Forecast. 15(2), 107–125 (1996)
42. Holt, C.C.: Forecasting seasonals and trends by exponentially weighted moving averages. Int. J. Forecast. 20(1), 5–10 (2004)
43. Berndt, D.J., Clifford, J.: Using dynamic time warping to find patterns in time series. In: KDD Workshop, vol. 10, pp. 359–370. Springer, Berlin (1994)
44. Huang, C., Zhang, C., Zhao, J., Wu, X., Yin, D., Chawla, N.: Mist: A multiview and multimodal spatial-temporal learning framework for citywide abnormal event forecasting. In: The World Wide Web Conference, pp. 717–728. International World Wide Web Conference Committee, Geneva (2019)
45. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778. IEEE, Piscataway (2016)
46. Liu, D., Hui, S., Li, L., Liu, Z., Zhang, Z.: A method for short-term traffic flow forecasting based on gcn-lstm. In: 2020 International Conference on Computer Vision, Image and Deep Learning (CVIDL), pp. 364–368. IEEE, Piscataway (2020)
47. Li, Z., Xiong, G., Chen, Y., Lv, Y., Hu, B., Zhu, F., Wang, F.-Y.: A hybrid deep learning approach with gcn and lstm for traffic flow prediction. In: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), pp. 1929–1933. IEEE, Piscataway (2019)
48. Huber, P.J.: Robust estimation of a location parameter. In: Breakthroughs in Statistics, pp. 492–518. Springer, Cham (1992)
49. Li, Y., Yu, R., Shahabi, C., Liu, Y.: Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In: International Conference on Learning Representations. ICML, San Diego (2018)
50. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems, vol. 27. MIT Press, Cambridge, MA (2014)
51. Tian, K., Guo, J., Ye, K., Xu, C.Z.: St-mgat: Spatial-temporal multi-head graph attention networks for traffic forecasting. In: 2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI), pp. 714–721. IEEE, Piscataway (2020)

How to cite this article: Huang, R., Chen, Z., Zhai, G., He, J., Chu, X.: Spatial-temporal correlation graph convolutional networks for traffic forecasting. IET Intell. Transp. Syst. 1–15 (2023). https://doi.org/10.1049/itr2.12330

1751956x, 0, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/itr2.12330 by University Of Sheffield, Wiley Online Library on [19/01/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License