International Journal of Data Warehousing and Mining
Volume 21 • Issue 1 • January-December 2025

Organizational Impact of Spatiotemporal Graph Convolution Networks for Mobile Communication Traffic Forecasting

Pan Ruifeng
https://orcid.org/0009-0004-3014-7270
Nanchang Vocational University, China

Mengsheng Wang
https://orcid.org/0009-0001-8567-5159
Nanchang Vocational University, China

Jindan Zhang
https://orcid.org/0000-0002-1499-8460
Xianyang Polytechnic Institute, China

Brij Gupta
https://orcid.org/0000-0003-4929-4698
Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan & Symbiosis Centre for Information Technology (SCIT), Symbiosis International University, Pune, India, & Center for Interdisciplinary Research, University of Petroleum and Energy Studies (UPES), Dehradun, India, & University of Economics and Human Science, Warsaw, Poland

Nadia Nedjah
https://orcid.org/0000-0002-1656-6397
Department of Electronics Engineering and Telecommunications, Brazil

ABSTRACT

Communication traffic prediction is of great guiding significance for communication planning management and improvement of communication service quality. However, due to the complex spatiotemporal correlation and uncertainty caused by the spatial topology and dynamic time characteristics of mobile communication networks, traffic prediction is facing enormous challenges. We propose a mobile traffic prediction method using dynamic spatiotemporal synchronous graph convolutional network (DSSGCN). DSSGCN has designed multiple components, which can effectively capture the heterogeneity in the local space-time map. More specifically, the network not only models the dynamic characteristics of nodes in the spatiotemporal graph of network traffic, but also captures the dynamic spatiotemporal characteristics of the edges of mobile service data with different time stamps. The outputs of these two components are fused by collaborative convolution to obtain the prediction results. Experiments on two ground truth mobile traffic datasets show that our DSSGCN model has good prediction performance.

KEYWORDS

Communication Traffic Prediction, GNN, GCN, Self-Attention Mechanism

DOI: 10.4018/IJDWM.368563

INTRODUCTION

With the rapid changes in population mobility, wireless communication systems have become increasingly complex, leading to many communication problems like network lag. Therefore, the development of advanced intelligent communication systems has become an urgent need. To achieve intelligent network management, the mobile communication traffic prediction has attracted widespread attention as a fundamental research problem in the spatiotemporal data mining of communication networks.

The main challenge in predicting mobile network traffic lies in effectively modeling the dynamic spatiotemporal features of communication traffic data.

For several decades, prediction methods have evolved from traditional statistical methods to modern deep learning techniques, promoting accuracy and broadening the application of data prediction across various fields. The autoregressive integrated moving average method, widely adopted for time dimension forecasting, captures autocorrelation in sequences (Box et al., 1976). Support vector machines identify the best separating hyperplane in high-dimensional space for classification and regression tasks, demonstrating good generalization ability (Cortes & Vapnik, 1995). Random forest constructs multiple decision trees and integrates their prediction results, making it widely used in classification and regression tasks. The long short-term memory network (LSTM) improves the recursive neural network, succeeding in catching long-term dependencies and performing well in time dimension prediction (Breiman, 2001; Hochreiter & Schmidhuber, 1997). The gated recurrent unit (GRU), which mirrors LSTM, offers a simpler structure and higher computational efficiency (Cho et al., 2014). The graph convolutional network (GCN) applies convolutional operations to graph-structured data, capturing spatial relationships between nodes (Kipf & Welling, 2017), which is significant in traffic flow prediction and social network analysis. Transformer processes sequence data through a self-attention mechanism (SAM), excelling at tasks that need to capture long-term dependencies in time dimension prediction (Vaswani et al., 2017). The time fusion transformer combines the SAM and sequence modeling, focusing on multi-view time dimension prediction. Hybrid models, with combine traditional statistical techniques with machine-learning algorithms, such as combining autoregressive integrated moving average with neural networks, have been developed to improve prediction accuracy (Lim et al., 2021; Zhang et al., 1998). However, such sequence learning models have significant shortcomings in high computational training.

In the past decade, graph neural networks (GNNs) have made remarkable achievements in handling graph-structured datasets, especially in the field of spatiotemporal prediction. Notable methods, such as spatiotemporal GCN (STGCN), Graph WaveNet, and dynamic graph convolutional neural network (CNN), have gained particular prominence (Wu et al., 2019; Yu et al., 2018; Zhang et al., 2019).

STGCN captures spatiotemporal correlation by extending convolution operations to graph structured data. These methods combine GCN and unidimensional CNN to process time dimension data and handle traffic flow prediction, achieving high prediction accuracy. Graph WaveNet, a graph-based deep learning model, captures long-range dependencies in graph data. It combines extended causal convolutions and graph convolutions to model long-term dependencies without increasing computational complexity. In addition, Graph WaveNet dynamically learns graph structures through adaptive graph convolution, improving its adaptability and generalization ability.

Dynamic GCN (DGCN) focuses on the processing of dynamic graph data, allowing it to handle graph structures that change over time. This method constructs dynamic graphs and applies graph convolution operations, capturing the temporal relationships between nodes, making it widely used in traffic flow prediction. Other methods, such as temporal GCN, combines GCN and GRU to catch spatiotemporal dependencies. Attention-based spatiotemporal GCN incorporates attention mechanisms to enhance the model's ability to capture important spatiotemporal features (Guo et al., 2019; Zhao et al., 2019).

These prediction methods and models continue to innovate and optimize within their respective application fields, promoting the forefront development of prediction technology.

In recent years, hybrid models combining GNNs and deep sequence models have made significant progress, particularly in wireless communication prediction. The attention-based hierarchical graph spatiotemporal model highlights the superiority of using hybrid graph-sequence models for temporal data prediction, especially by combining GNN with sequence models (Behrouz et al., 2024). The data-driven intelligent control for wireless communication network explores the latest application of data-driven intelligent control to optimize wireless communication networks using big data and artificial intelligence (Huo et al., 2024). Joint statistical modeling and machine learning focuses on achieving efficient traffic prediction in network management through joint statistical modeling and machine learning methods (Lo Schiavo et al., 2022).

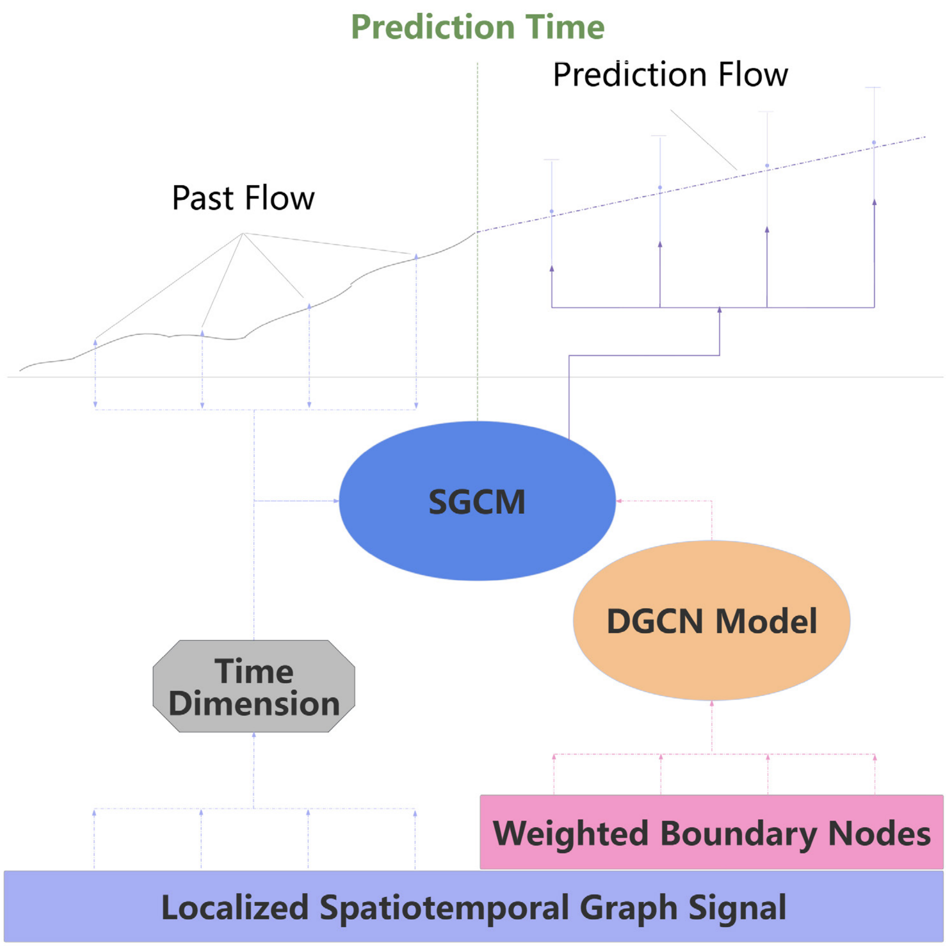
These studies demonstrate the feasibility and effectiveness of combining GNNs and sequence models for traffic prediction. They also provide background support for selecting the technical path and framework design for communication traffic prediction in this study, offering important inspiration and a theoretical basis for the multi-model fusion strategy proposed here.

Although GCN performs well in mobile communication network traffic prediction, there are still shortcomings. GCN captures node features in static graph structures; however, it struggles to capture dynamic changes over time. As a result of the high dynamism and temporal variability of mobile communication network traffic datasets, GCN cannot effectively capture the dynamic features that change over time, reducing forecast accuracy. Additionally, as the network size grows, the computational complexity of GCN significantly increases, leading to higher demands on computing resources and time consumption. This poses challenges for real-time traffic prediction in large-scale mobile communication networks.

Traditional GCN focuses on the local neighborhood information of nodes, ignoring the dependency relationships between distant nodes. However, in mobile communication networks, the traffic dependency relationship between long-distance base stations is equally important. This local structural limitation can negatively affect prediction performance. Furthermore, mobile communication networks may suffer from sparsity issues, where data is lacking in certain time periods or in specific regions. GCN performs poorly in handling sparse data, which can lead to inaccurate predictions.

To settle these issues, a dynamic spatiotemporal synchronous GCN (DSSGCN) for mobile communication network flow forecasting has been proposed. Figure 1 shows that the core concept of DSSGCN is the use of DGCN to introduce the time dimension, capture dynamic change features, and combine it with SAM, which strengthens the ability of the model to catch long-term correlations. Weighted boundary nodes are used to improve the model's ability to handle sparse data, while a spatiotemporal synchronous graph convolution module (SGCM) was constructed to capture complex spatiotemporal correlations within local spatiotemporal graphs. By aggregating long-range spatiotemporal correlations and heterogeneity, the mobile communication network traffic prediction can be achieved. Compared with current GCN-based methods, this model achieves significant improvements, as validated in the prediction experiments in the following sections.

Figure 1. Main components of dynamic spatiotemporal synchronous graph convolutional network



Note. SGCM = synchronous graph convolution module; DGCN = dynamic graph convolutional network.

The main contributions of this article are as follows:

- A new traffic prediction framework based on DGCN for mobile communication networks is proposed. This is characterized by synchronously capturing local spatiotemporal correlations rather than using different types of independent neural network modules.
- The proposed DSSGCN constructs a spatiotemporal SGCM to capture correlations and heterogeneity in long-range spatiotemporal graphs. Deploying this module in each period of long-range spatiotemporal communication reveals the potential spatiotemporal correlations of more complex dynamic network communication systems.

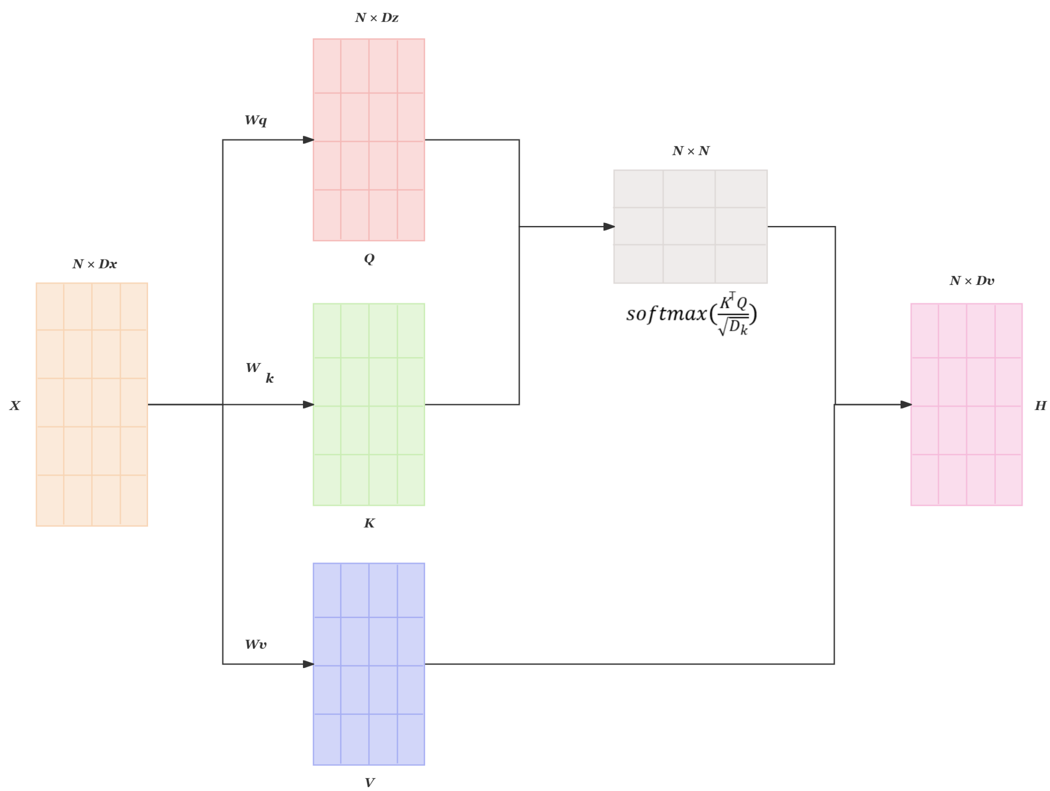
Additionally, the adaptive GCN approach models dynamic data flows, which is crucial for managing and optimizing data in modern data warehouses. This method enhances the ability to capture complex relationships within data, such as spatial and temporal dependencies. This is essential for improving the efficiency and accuracy of data mining processes.

Furthermore, the approach facilitates more effective data storage, retrieval, and processing in data warehousing. By leveraging dynamic graph structures, this study enables the system to handle evolving data streams, making it well-suited for real-time analytics and predictive modeling, where the ability to manage and analyze massive, constantly changing datasets is paramount.

RELATED WORK

Traditional CNNs perform well in handling grid-like data. However, there are limitations when dealing with non-Euclidean structured graph data. In response to these shortcomings, researchers have proposed various improvement methods, including the introduction of SAM (see Figure 2), the use of extended convolutional techniques, and the construction of dynamic graph models. These improvement methods enhance the functionality of the models, improving the feasibility and effectiveness of practical applications.

Figure 2. Self-attention mechanism calculation process



Convolutional-Based Methods

The following are convolutional-based spatiotemporal network modeling methods:

- **GCN:** The semi-supervised GCN model captures local spatial dependencies between nodes by applying convolution calculations to neighboring nodes within the graph.

- **STGCN:** This captures spatial correlations through GCN and temporal dependencies through unidimensional CNN, addressing the shortcomings of static GCN in processing time-varying data.
- **Graph WaveNet:** This model captures long-distance dependencies without significantly increasing computational complexity. It also enhances the processing ability of dynamic spatiotemporal data by dynamically learning graph structures.
- **DGCN:** This captures dynamic relationships between nodes through dynamic graph construction and graph convolution operations. It is widely used in social network analysis and traffic flow prediction, demonstrating strong predictive capabilities.
- **SAM:** This method is introduced into graph convolutional models to improve the ability to extract long-term correlations. It significantly enhances the capacity to capture important spatiotemporal features.

Problem Formulation

Wireless network traffic is highly dynamic, influenced by factors like changes in network nodes, fluctuations in user behaviors, and environmental interferences. Due to the complexity and variability of these factors, accurately predicting future traffic patterns is a challenging problem. This study aims to use historical traffic data to predict network traffic for a certain time. This prediction is crucial for resource allocation, traffic scheduling, and network performance optimization. Accurate traffic forecasting helps to reduce congestion, ensure quality of service, and optimize bandwidth and computational resource distribution.

Traffic data in wireless networks typically exhibits strong spatiotemporal dependencies, where traffic changes over time and is influenced by interactions between network nodes. The dynamic nature of these networks poses challenges for traditional forecasting methods, making it difficult to capture these complex dependencies and leading to inaccurate predictions. This study defines the problem as predicting the communication traffic of wireless network nodes for a future time based on historical traffic data, while accounting for both spatial and temporal characteristics. To solve this problem, the study aims to develop a model capable of dynamically capturing these spatiotemporal dependencies and accurately predicting future traffic trends.

GCN exhibit excellent performance in handling non-Euclidean structured data. The following are specific steps and key formulas involved in the preparation of graph convolution.

Improving Set Response Rates

Each base station or network unit in a mobile communication network is defined as a node. The connection correlation is defined as edges. The weights of edges are defined based on geographic distance or communication strength. The node connectivity is denoted by constructing an adjacent matrix, where each element represents an edge weight:

$$\mathscr{A}_{ij} = \begin{cases} w_{ij} \; \textit{Node i and j have connection correlation} \\ \qquad\qquad 0 \; \textit{Other situations} \end{cases} \tag{1}$$

where $\mathscr{A}$ denotes the correlation of all node connections, $\mathscr{A}_{ij}$ indicates the edge weight of node i and j, and w_{ij} is a weight based on geographic distance or communication strength.

Graph Signal and Convolution

The characteristics of each base station node include historical traffic data, temporal characteristics, spatial characteristics, and more. They form the node feature matrix $\mathscr{X} \in \mathbb{R}^{N\times F}$, where N denotes node amount and F denotes characteristic dimension. The characteristics of edges convolution include physical distance between base stations, communication strength, and more. The edge feature matrix is denoted as $\mathscr{E} \in \mathbb{R}^{M \times F_e}$, where $\mathscr{M}$ denotes edge amount and F_e denotes edge characteristic dimension.

Graph convolution operations use edge features to weight the adjacent matrix and improve the information propagation process:

$$\mathscr{H}^{(l+1)} = \sigma\left(\widetilde{\mathscr{R}}^{-1/2}\widetilde{\mathscr{A}}\left(\mathscr{E}\right)\widetilde{\mathscr{R}}^{1/2}\mathscr{H}^{l}\mathscr{W}^{l}\right) \tag{2}$$

in which $\widetilde{\mathscr{A}} = \mathscr{A} + I$ denotes weighted adjacent matrix obtained by adding a unit matrix and combining it with edge features matrix $\mathscr{E}$, $\widetilde{\mathscr{R}}$ denotes degree matrix, $\widetilde{\mathscr{R}}_{ii} = \sum_j \widetilde{\mathscr{A}}_{ij}$, $\mathscr{H}^{l}$ and $\mathscr{W}^{l}$ denotes l-th layer's node feature matrix and weight matrix, respectively, and σ denotes activation function (such as *ReLU*).

In spatiotemporal convolutional networks (such as STGCN), combining graph convolution and time convolution to capture spatiotemporal correlations:

$$\mathscr{H}^{(t+1)} = GCN\left(\mathscr{X}^{t}\right) * f_t \tag{3}$$

where * represents convolution operations in temporal dimension and f_t denotes time convolution kernel.

Attention Mechanism and Loss Function

In attention-based GNNs (Velickovic et al., 2018), edge features can be used to calculate attention weights between nodes:

$$e_{ij} = LeakyReLU\left(a^{T}\left[\mathscr{W} h_i \parallel \mathscr{W} h_j \parallel \mathscr{E}_{ij}\right]\right) \tag{4}$$

in which a denotes attention weight vector, h_i and h_j are the features of nodes i and *j*, and $\mathscr{E}_{ij}$ denotes feature of edges *i~j*.

In terms of model training and validation, this study chooses the mean absolute error (MAE) (Willmott & Matsuura, 2005) as the loss function:

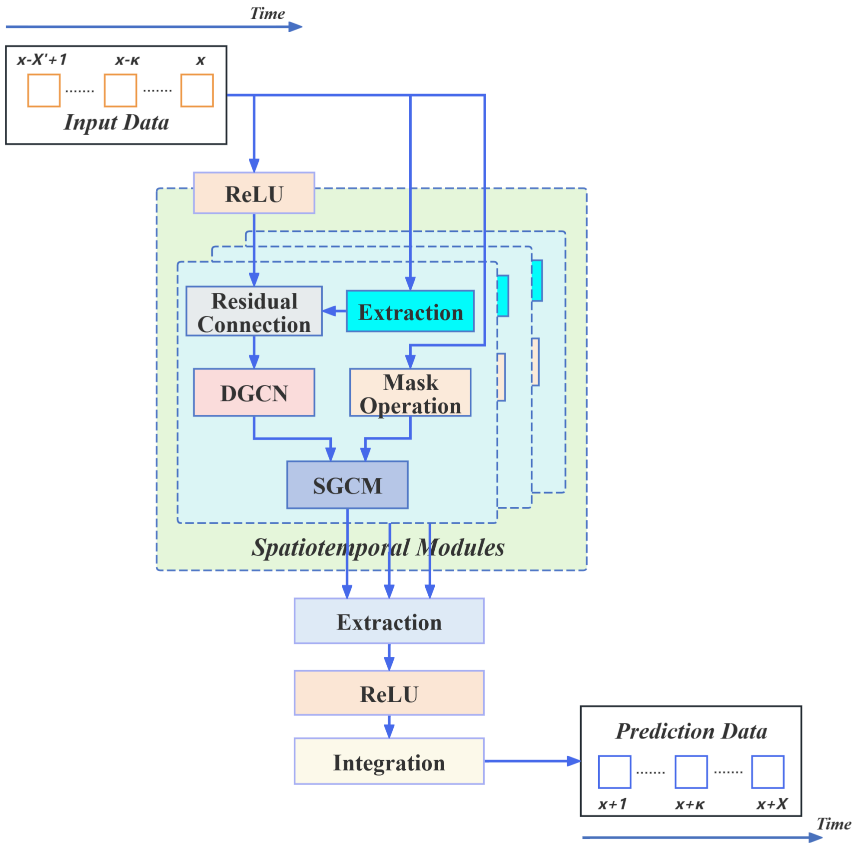
$$\mathscr{L} = \frac{1}{N}\sum_{i=1}^{N}\left|\mathscr{Y}_i - \widehat{\mathscr{Y}}_i\right| \tag{5}$$

in which N is sample size, $\mathscr{Y}_i$ is i-th specimen actual data, $\widehat{\mathscr{Y}}_i$ is i-th specimen prediction data, and $\left|\mathscr{Y}_i - \widehat{\mathscr{Y}}_i\right|$ denotes the absolute error between i-th specimen prediction data and actual data.

DSSGCN PREDICTIVE MODEL

Figure 3 illustrates the study's proposed DSSGCN framework. First, the entry cellular communication network flow is handled by a modified linearity module. Then, it is fed into a superposed spatiotemporal module (STM), which extracts complex spatiotemporal features from the input features. The STM performs feature fusion and aggregation through a gating mechanism, attention mechanism, and multi-level dynamic graph convolution. Finally, the extracted features are mapped to the prediction results through multiple output layers.

Figure 3. Dynamic spatiotemporal synchronous GCN (DSSGCN) architecture for mobile communication traffic forecasting



Note. ReLU = rectified linear unit; SGCM = synchronous graph convolution module; DGCN = dynamic graph convolutional network.

Construct Adjacent Matrix

The model constructs an adjacent matrix through adaptive adjacent matrix and spatial attention mechanism, as shown in Formula 1, to dynamically adjust the weights between nodes and capture complex spatiotemporal relationships (Li et al., 2018). The initialized node vector and boundary vector adaptive parameters are used to adaptively generate the adjacent matrix. In the forward propagation process, the weights between nodes are dynamically adjusted by calculating the self-adaption adjacent matrix. Specifically, the matrix is generated through the product of node vectors and activation functions. The calculation process is as follows:

$$\mathscr{H} = \mathscr{H}_1 \bullet \mathscr{H}_2 \tag{6}$$

$$\mathscr{R} = ReLU(\mathscr{H}) = max(0, \mathscr{H}) \tag{7}$$

$$\mathscr{A}_{ij} = \frac{exp\left(\mathscr{R}_{ij}\right)}{\sum_{k=1}^{N} exp\left(\mathscr{R}_{ik}\right)} \tag{8}$$

In Formula 6, the node vector matrix $\mathscr{H}_1$ has a dimension of (N, d), and is the embedding vector for each node. The node vector matrix $\mathscr{H}_2$, with dimensions (d, N), is a weight matrix used to calculate the adaptive adjacent matrix, where d denotes feature dimension. The result of the matrix dimension multiplication, $\mathscr{H}$, is (N, N).

In Formula 7, $\mathscr{R}$ is the matrix activated by the activation function rectified linear unit, or ReLU. Formula 8 applies the softmax function, which transforms each row of input $\mathscr{R}$ into a probability distribution. This process calculates the similarity between nodes, converts it into a probability distribution through the softmax function, and forms an adaptive adjacent matrix $\mathscr{A}_{ij}$.

In addition, the study uses spatial attention mechanism to further regulate the adjacent matrix (Vaswani et al., 2017; Wu et al., 2020). This mechanism, which is based on input features, calculates the correlation between nodes through a series of parameters. It can be formulated as follows:

$$lhs = \mathscr{X} \bullet \mathscr{W}_1 \bullet \mathscr{W}_3 \tag{9}$$

$$rhs = \mathscr{X} \bullet \mathscr{W}_2 \bullet \mathscr{W}_4 \tag{10}$$

$$sum = cat[lhs[:, idx, :], rhs[:, idy, :]] \tag{11}$$

$$\mathscr{S}_{n,c_{out},h,w} = \sum_{c_{in}=0}^{c_{in}-1} \sum_{i=0}^{k_h-1} \sum_{j=0}^{k_w-1} \mathscr{W}_{c_{out},c_{in},i,j} \bullet sum_{n,c_{in},h+i,w+j} + b_{c_{out}} \tag{12}$$

where $\mathscr{W}_1$ and $\mathscr{W}_2$ are parameters used for weighting time-step features. $\mathscr{W}_2$ and $\mathscr{W}_4$ are parameters used for dimensionality reduction; $\mathscr{X}$ is used as the input feature matrix. These are weighted and transformed by the parameter matrices during the forward propagation process. The feature map sum is concatenated by the left operand lhs and the right operand rhs, while idx and idy are indexed arrays used to select specific vertices. $\mathscr{W}_{c_{out},c_{in},i,j}$ denotes convolution kernel weight, $b_{c_{out}}$ denotes convolution bias, $\mathscr{S}_{n,c_{out},h,w}$ is the generated new adjacent matrix, $n \in \{0, 1, 2, ..., N-1\}$ is the batch dimension index, c_{out} is the output channel dimension index, and c_{in} is the input channel dimension index. h and w denote the output characteristic chart's height and width dimension indexes. i and j are height and width dimension indexes of the convolutional kernel.

This new adjacent matrix and adaptive adjacent matrix form the final adjacent matrix $\mathscr{A}$.

STM

The constructed STM model improves prediction accuracy by capturing the temporal and spatial dependencies of nodes. Specifically, the first-generated adjacent matrix represents the spatial relationships between nodes. Spatial features are then extracted using graph convolution operations, while time features are captured through time convolution or recurrent neural networks. Afterwards, the spatiotemporal features are fused.

The SGCM module uses stacked multilayer graph convolutional to further extract spatiotemporal features. By using an adaptive adjacent matrix and a spatial attention mechanism, the weights between nodes are dynamically adjusted to capture more important spatiotemporal dependencies. Finally, the outputs of all convolutional layers are integrated to generate the prediction result. This process integrates the strengths of graph and time convolution, allowing more effective processing of spatiotemporal data.

Graph Convolutional Layer

In the specific implementation, there are also auxiliary modules, such as graph convolutional layers, linear layers, and multi-layer perceptrons. These are combined into a dynamic GCN module to process node and boundary features. The formulas are listed here:

$$\mathscr{H}^{'} = \mathscr{A} \bullet \mathscr{H} \tag{13}$$

$$\widehat{\mathscr{A}} = \widetilde{\mathscr{R}}^{-\frac{1}{2}} \widetilde{\mathscr{A}} \widetilde{\mathscr{R}}^{-\frac{1}{2}} \tag{14}$$

$$\hbar^{(l+1)} = \sigma\left(\widehat{\mathscr{A}}\, \hbar^{l} \mathscr{W}^{l} + b^{l}\right) \tag{15}$$

$$\mathscr{H}^{l} = \sigma(\mathscr{W}^{l} \mathscr{H}^{(l-1)} + b^{l}) \tag{16}$$

where $\hbar^{1}$ denotes the input characteristic matrix of layer l, $\widehat{\mathscr{A}}$ denotes normalized adjacent matrix, and b^{l} is the bias vector of the l-th layer.

The aggregation of node features based on the graph adjacent matrix is a core concept of graph convolutional layers, aiming to catch dependency relationships between nodes. This study used two forms of graph convolution operations within the framework.

The first form employed the Einstein summation convention, with the new node feature matrix $\mathscr{H}^{'}$ obtained through Formula 13. The second form transposed the adjacent matrix for feature aggregation, as defined in Formula 15.

The linear layer implemented standard two-dimensional convolution operations to perform linear transformations on node features. This layer contained a two-dimensional convolution module, calculated using Formula 16. Additionally, another layer used convolution kernels of different sizes and dilation rates to consider different contextual information during feature extraction.

Overall, these modules achieve feature aggregation and extraction of graph structured data through graph convolution operations, providing a foundation for subsequent feature fusion and prediction. These implementation ideas combine the classic concept of graph CNN, capturing the relationships between nodes through convolutional operations and effectively processing spatiotemporal graph data.

Time Convolutional Layer

The time convolutional layer performs convolution operations on temporal dimension data. This is mainly reflected in temporal dimension features extraction and aggregation (Yan et al., 2018). Time convolutional layers aim to process time dimension data through one-dimensional convolution operations, capturing temporal dependencies and dynamic patterns of change (Dai & Zhong, 2020). In this study, the time convolution layer is usually implemented through a two-dimensional convolution module, which is configured to perform convolution operations in the time dimension. The formula is as follows:

$$\mathscr{H}_{bktn} = \sigma\left(\sum_{d=0}^{D-1} \sum_{t_k=0}^{T-1} \mathscr{W}_{kdt_k,0}\, \mathscr{X}_{b,d,t+t_k,n} + b_k\right) \tag{17}$$

where $\mathscr{H}_{\mathrm{bktn}}$ is the value of the convolutional output feature map on the b-th sample, k-th channel, t-th time step, and n-th node. $\sum_{\mathrm{c=0}}^{\mathrm{C-1}}$is the sum of the input channels, $\sum_{\mathrm{t_k=0}}^{\mathrm{T-1}}$is the sum of the time dimension of the convolutional kernel, $\mathscr{W}_{\mathrm{kdt_k,0}}$ is the weight of the k-th output channel, d-th input channel, t-th time step in the convolutional kernel, $\mathscr{X}_{\mathrm{b,c,t+t_k,n}}$ is the value of the b-th sample, c-th channel, $\mathrm{t} + \mathrm{t_k}$ time step, and n-th node in the input tensor.

Specifically, a sequence characteristic matrix comprising multiple time steps forms the input of time convolutional layer. By setting appropriate convolution kernel size and step size, time convolutional layers slide along the time dimension and perform convolution calculations on features at each time step.

The convolution operation involves applying convolution kernels to local windows within the time dimension to extract features from within that window. This process converts the original temporal dimension data into higher-order characteristic representations, allowing the model to better capture vibrant changes and correlations in time. These higher-order feature representations are then passed to the graph convolutional layer and other network layers for further feature extraction and prediction tasks.

SGCM

The core concept of SGCM is to capture complex correlations in spatiotemporal information by combining graph convolution and time convolution (Zhang et al., 2020). In spatiotemporal data, spatial and temporal relationships are often intertwined, making it essential to consider both simultaneously for effective modeling.

SGC uses GCN to model node relationships in space. Graph convolution can capture spatial dependencies in graph structures and learn spatial feature representations of each node through an adjacent matrix for message passing and aggregation. Specifically, the input feature matrix is transformed through graph convolutional layers to generate spatial feature representations. This step is represented as follows:

$$\mathscr{H}_s^{(l+1)} = \sigma\left(\widetilde{\mathscr{A}}\,\mathscr{H}_s^l \mathscr{W}_s^l\right) \tag{18}$$

where $\mathscr{H}_s^l$ and $\mathscr{W}_s^l$ denote l-th layer's node characteristic matrix and weight matrix, respectively.

Temporal convolution utilizes one-dimensional convolution to extract temporal features. It captures dynamic changes within the time dimension and models time dependencies by performing convolution operations on the time dimension. Specifically, spatial feature representations are transformed through temporal convolutional layers to generate spatiotemporal feature representations. This step can be represented as follows:

$$\mathscr{H}_t^{(l+1)} = \sigma\left(\mathscr{W}_t * \mathscr{H}_t^l + b_t\right) \tag{19}$$

where $\mathscr{W}_t$ denotes the time convolutional layer weight, $\mathscr{H}_t^l$ denotes time feature matrix of the l-th layer, and * is a two-dimensional convolution according to Formula 17.

Spatiotemporal feature fusion achieves joint modeling of spatiotemporal features by alternately applying graph convolution and time convolution layers. Through multi-level spatiotemporal feature extraction, complex spatiotemporal dependencies can be captured. Specifically, the spatiotemporal feature representation of each layer represented as follows:

$$\mathscr{H}^{(l+1)} = \sigma\left(\mathscr{W}_{ST} * \mathscr{H}^l + b_{ST}\right) \tag{20}$$

where $\mathscr{W}_{ST}$ and b_{ST} denote spatiotemporal convolutional layer weights and bias terms, respectively.

EXPERIMENTS

To assess the function of DSSGCN, this study performs experimentation and comparisons using two ground-truth mobile communication flow information from the Big Data Challenge initiated by Telecom Italia (Barlacchi et al., 2015).

Datasets

The study evaluated DSSGCN using the Milan Regional Public Mobile Traffic Dataset released by Telecom Italia. The Milan region consists of 1,000 grids, each covering an area of approximately 235x235 meters. The data was collected over two months, from November 1, 2013, to January 1,

2014. Communication flow was recorded every 10 minutes. To address data sparsity issues, the traffic sampling rate was adjusted to one-hour intervals.

According to time density, the datasets contain 207 and 400 traffic points, respectively. These datasets are divided into training, testing, and validation sets in a ratio of 7:2:1. The adjacent matrix of the node graph is constructed from the spatial dimension of the grid using a threshold Gaussian kernel (Shuman et al., 2013). Table 1 presents detailed statistical information on the datasets.

Table 1. Statistical figures of Milan and Milan400

Datasets	# Traffic Points
Milan	207
Milan400	400

Experiment Settings

In these experiments, the study used 24-hour historical data to forecast the next hour's communication flow. Specifically, the input dimension was 24, and the prediction step was 1. Self-adaption matrices were inserted with equidistribution arguments of size 10. All experiments were repeated five times. The learning rate was set to 0.001, with a learning rate decline of 0.97. The batch size was 8, and both the remaining and expansion channels were set to 60. The dropout rate to 0.3, and the weight decay rate was 0.0001. The optimization models consist of three STMs, with each SGCM containing three graph convolution operations and 40 intermediate filters. All experiments were conducted using a computer equipped with an Nvidia GeForce RTX 1070 GPU.

Baseline Methods

The study contrasted its DSSGCN approach with different communication flow forecasting methods, including GRU, recurrent neural network, LSTM, multi-range attentive bicomponent GCN, dual dynamic spatiotemporal GCN; and Graph WaveNet (Chen et al., 2020; Sun et al., 2022; Sutskever et al., 2014; Zhou & Nelson, 2002). The study used three error metrics to compare the performance of each method: MAE, mean absolute percentage errors (MAPE), and root mean squared errors (RMSE).

Experiment Results

Comparison and Analysis

The study contrasted the predictive function of DSSGCN with six base-line methods at 12 prediction levels on two datasets (see Table 2). The STSGCN consistently outperformed other base-line approaches on both datasets.

Table 2. Prediction results using various approaches on Milan and Milan400

Methods	Dataset	Milan			Milan400		
	Metrics	MAE	RMSE	MAPE (%)	MAE	RMSE	MAPE (%)
RNN		0.0221	0.0307	104.4	0.0237	0.0368	123.74
LSTM*		0.0202	0.0295	68.99	0.0223	0.0353	105.15
GRU		0.0196	0.0287	65.54	0.0224	0.0355	106.85
MRA-BGCN		0.0219	0.0306	98.77	0.0239	0.0367	127.2

continued on following page

Table 2. Continued

Methods	Dataset	Milan			Milan400		
	Metrics	MAE	RMSE	MAPE (%)	MAE	RMSE	MAPE (%)
Graph WaveNet		0.0198	0.0329	19.31	0.0245	0.0397	17.29
DDSTGCN		0.0194	0.0308	14.05	0.0222	0.0371	18.65
DSGCN (study)		0.0171	0.0271	13.73	0.0197	0.0329	18.93
DSSGCN (study)		0.0115	0.0198	8.67	0.0142	0.0271	12.76

Note. * LSTM, or long short-term memory, is a type of recurrent neural network that captures long-term dependencies in sequential data. MAE = Mean Absolute Error; MAPE = Mean Absolute Percentage Errors; RMSE = Root Mean Squared Errors; RNN = Recurrent Neural Network; GRU = Gated Recurrent Unit; MRA-BGCN = Multi-Range Attentive Bicomponent GCN; DDSTGCN = Dual Dynamic Spatial-Temporal GCN; DSGCN = Dynamic Spatiotemporal GCN; DSSGCN = Dynamic Spatiotemporal Synchronous GCN.

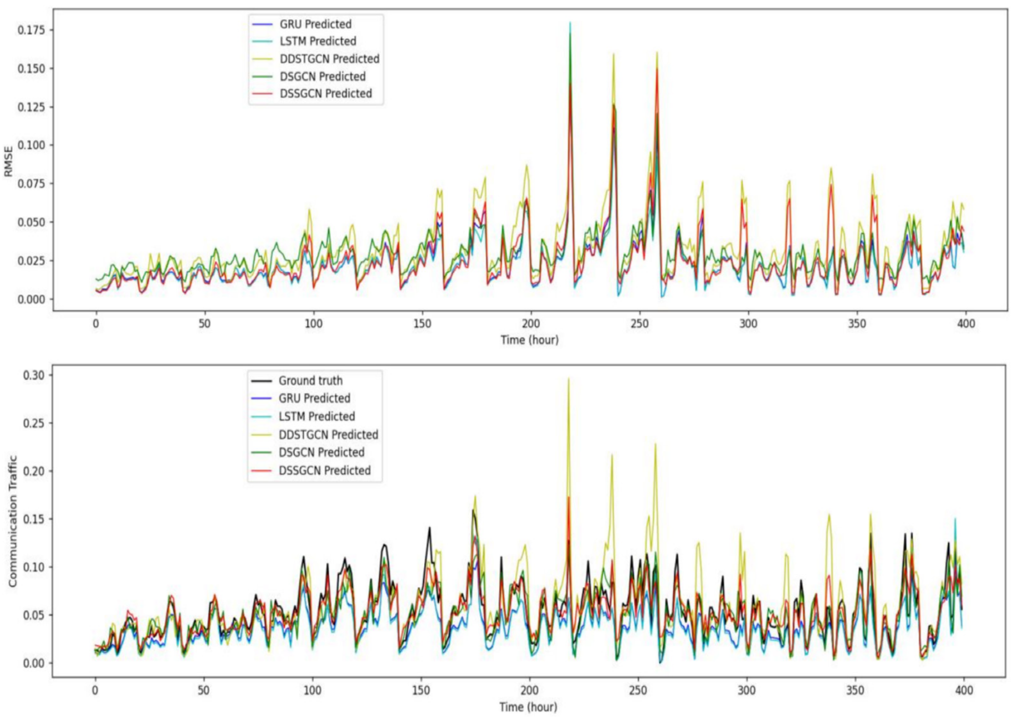
RNN, LSTM, and GRU models only consider temporal correlation and do not use the spatial correlations in spatiotemporal network. MRA-BGCN and Graph WaveNet utilize two modules to model spatial-temporal correlation, with a shared module at different times to extract long-term spatiotemporal correlation. However, they ignore the heterogeneity of spatiotemporal network data. DDSTGCN considers the dynamic hypergraph mechanism, outperforming the baseline on the Milan dataset. However, in the Milan400 dataset, the performance in terms of MAPE is not as good as that of Graph WaveNet. DSGCN optimizes the SAM and obtains better capability on the Milan dataset without the SGCM. Yet, on the Milan400 dataset, MAPE is slightly higher than DDSTGCN. Based on DSGCN, the DSSGCN combines the SGCM, considers local spatiotemporal correspondence, and captures spatiotemporal heterogeneity, obtaining better results than the most advanced previous models. This showcases the ascendancy of this study's method in characterizing the spatiotemporal features of cellular network communication flow.

Visualization of Results

To visually demonstrate the test results, the study compared the average RMSE and communication traffic prediction results of five methods on the Milan400 dataset, as shown in Figure 4. All five methods forecast the communication traffic peak-valley. Contrasted with other models, the DSSGCN model displays better overall performance in terms of prediction.

In Figure 4(a), the DSSGCN model curve is lower than the others for most of the time, indicating that DSSGCN performs better than other methods. In Figure 6(b), the DSSGCN method performs better in traffic prediction, with the DSSGCN (red curve) closely following the ground truth communication traffic (black thick curve). This indicates that DSSGCN outperforms other methods in communication traffic prediction. Notably, in long-term prediction, DSSGCN consistently produces more accurate results, highlighting that the strategy based on synchronous graph convolution combined with SAM can better explore the dynamic spatial-temporal patterns of communication flow.

Figure 4. Performance of various models on Milan400 (a) average forecast results using various models on Milan400 under RMSE (b) forecast results of communication traffic using various models on Milan400



Note. GRU = Gated Recurrent Unit; LSTM = Long Short-Term Memory; DSGCN = Dynamic Spatiotemporal GCN; DSSGCN = Dynamic Spatiotemporal Synchronous GCN; RMSE = Root Mean Squared Errors; DDSTGCN = Dual Dynamic Spatial-Temporal GCN.

To further demonstrate the predictive performance, the study arranged the actual communication traffic, predicted communication traffic, prediction error, and standard deviation of prediction results in a square matrix. Figure 5(a) and Figure 5(b) represent the level of communication traffic with different colors. Darker colors indicate greater communication traffic. The figure shows the contrast of forecast results with the ground-truth situation, indicating that the predicted communication traffic is similar to the actual communication traffic.

Figure 5(c) shows that the overall prediction error values are relatively small, with most errors ranging from 0.001 to 0.01. This indicates that the model's prediction results are relatively accurate in most cases. According to Figure 5(d), most of the standard deviations fall between 0.001 and 0.01, indicating that the model's predictions are relatively stable in most cases. However, some base stations (like columns 6 and 7) have large errors and standard deviations at multiple time steps. Thus, the traffic of these base stations may have significant volatility or complexity, making the method unable make firm predictions. Therefore, prediction results will be unstable.

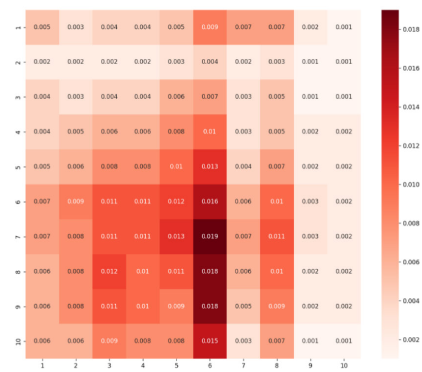
It is possible to introduce more complex models or enhance the robustness of the model to address situations with large traffic fluctuations.

Figure 5. Visual display of predictive performance on Milan400 (a) ground truth communication traffic (b) prediction of communication traffic (c) error between predicted and actual values (d) standard deviation of predicted results

(a) Ground truth communication traffic

(b) Prediction of communication traffic

(c) Error between predicted and actual values



(d) Standard deviation of predicted results

Computational Complexity

Using the Milan and Milan400 datasets, the study further compared the training time and inference time of the DSSGCN model with other methods. Table 3 reports the experimental results.

On the Milan dataset, the DSSGCN proposed has similar training and inference time compared to DDSTGCN. However, on the Milan400 dataset, the training and inference of DSSGCN are longer than DDSTGCN, indicating that their computational complexity remains the same at low data densities.

The proposed DSSGCN method has higher computational complexity at high data densities due to the inclusion of layered SGCM modules. On both datasets, the training and inference times are slightly longer compared with LSTM and GRU, primarily because of the added self-attention module. However, considering the significant performance enhancement of the proposed DSSGCN method, the relatively high computational cost is justifiable, and the small time difference is acceptable.

Table 3. Computational complexity on Milan and Milan400

Dataset	Computation Time (Training Time (secs/epoch)/Inference Time (secs))				
	LSTM	MRA-BGCN	GRU	DDSTGCN	DSSGCN(ours)
Milan	6.6774/0.859	196.5487/17.6722	5.5709/0.8118	17.7491/2.1783	14.8003/1.5889
Milan400	5.5806/0.8161	66.5813/8.8921	3.0408/0.3569	9.4044/1.0977	26.3928/3.1031

Note. LSTM = Long Short-Term Memory; MRA-BGCN = Multi-Range Attentive Bicomponent GCN; GRU = Gated Recurrent Unit; DDSTGCN = Dual Dynamic Spatial-Temporal GCN; DSSGCN = Dynamic Spatiotemporal Synchronous GCN.

CONCLUSION

This article presents a deep learning-based DSSGCN for predicting traffic in cellular wireless communication networks. This model combines GCN and gated linear units to improve prediction accuracy by capturing spatiotemporal dependencies. The core of this model consists of a SGCM and multiple GCN layers. Each GCN layer captures the spatiotemporal dependencies between nodes through a dynamic adjacent matrix, while filtering important information through a gating mechanism. Additionally, the model includes several attention mechanisms that dynamically adjust the edge weights of the graph, enhancing its ability to learn complex spatiotemporal patterns.

This study introduces an in-depth analysis of prediction variability, focusing on standard deviation patterns across spatial and temporal dimensions. By exploring the variability in forecast errors, the study provides valuable insights for improving forecasting model stability and robustness—key for optimizing end-user computing applications in dynamic network environments. The findings contribute to refining predictive models, enabling more accurate and reliable network traffic management. Overall, the DSSGCN model implemented in this article provides an efficient, flexible, and accurate solution for traffic prediction tasks in cellular wireless communication networks.

COMPETING INTERESTS

The authors of this publication declare there are no competing interests.

FUNDING

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors. Funding for this research was covered by the authors of the article.

PROCESS DATES

01, 2025

This manuscript was initially received for consideration for the journal on 11/24/2024, revisions were received for the manuscript following the double-anonymized peer review on 12/24/2024, the manuscript was formally accepted on 01/15/2025, and the manuscript was finalized for publication on 01/22/2025

CORRESPONDING AUTHOR

Correspondence should be addressed to Pan Ruifeng; panruifeng@nvu.edu.cn

REFERENCES

Barlacchi, G., De Nadai, M., Larcher, R., Casella, A., Chitic, C., Torrisi, G., Antonelli, F., Vespignani, A., Pentland, A., & Lepri, B. (2015). A multi-source dataset of urban life in the city of Milan and the province of Trentino. *Scientific Data*, *2*(1), 150055. DOI: 10.1038/sdata.2015.55 PMID: 26528394

Behrouz, A., Parviz, A., Karami, M., Sanford, C., Perozzi, B., & Mirrokni, V. (2024). Best of both worlds: Advantages of hybrid graph sequence models.

Box, G. E. P., Jenkins, G. M., & Reinsel, G. C. (1994). *Time series analysis: Forecasting and control* (3rd ed.). Prentice Hall.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. DOI: 10.1023/A:1010933404324

Chen, W., Chen, L., Xie, Y., Cao, W., & Feng, X. (2020). Multi-range attentive bicomponent graph convolutional network for traffic forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(4), 3529–3536. DOI: 10.1609/aaai.v34i04.5758 PMID: 32944409

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*. DOI: 10.3115/v1/D14-1179

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. DOI: 10.1007/BF00994018

Dai, Z., & Zhong, Z. (2020). Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(01), 507–514.

Guo, S., Lin, Y., Feng, N., Song, C., & Wan, H. (2019). Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*(01), 922–929. DOI: 10.1609/aaai.v33i01.3301922

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. DOI: 10.1162/neco.1997.9.8.1735 PMID: 9377276

Huo, W., Yang, H., Yang, N., Yang, Z., Zhang, J., & Nan, F. (2024). Recent advances in data-driven intelligent control for wireless communication: A comprehensive survey. *arXiv preprint arXiv:2408.02943*.

Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2018). Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations (ICLR 18).*

Lim, B., Arik, S. O., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, *37*(4), 1748–1764. DOI: 10.1016/j.ijforecast.2021.03.012

Lo Schiavo, L., Fiore, M., Gramaglia, M., Banchs, A., & Costa-Perez, A. (2022). Forecasting for network management with joint statistical modelling and machine learning. *IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)* (pp. 60–69). DOI: 10.1109/WoWMoM54355.2022.00028

Shuman, D. I., Narang, S. K., Frossard, P., Ortega, A., & Vandergheynst, P. (2013). The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, *30*(3), 83–98. DOI: 10.1109/MSP.2012.2235192

Sun, Y., Jiang, X., Hu, Y., Duan, F., Guo, K., Wang, B., Gao, J., & Yin, B. (2022). Dual dynamic spatial-temporal graph convolution network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, *23*(12), 23680–23693. DOI: 10.1109/TITS.2022.3208943

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). *Sequence to sequence learning with neural networks. Proceedings Adv. Neural Inf. Process. Syst* (*Vol. 27*). NIPS.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2018). Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research*, *30*(1), 79–82. DOI: 10.3354/cr030079

Wu, Z., Pan, S., Long, G., Jiang, J., Chang, X., & Zhang, C. (2019). Graph WaveNet for deep spatial-temporal graph modeling. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI'19).* DOI: 10.24963/ijcai.2019/264

Wu, Z., Pan, S., Long, G., Jiang, J., Chang, X., & Zhang, C. (2020). Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD 20).* DOI: 10.1145/3394486.3403118

Yan, S., Xiong, Y., & Lin, D. (2018). ST-GCN: Spatial temporal graph convolutional networks for skeleton-based action recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1). Advance online publication. DOI: 10.1609/aaai.v32i1.12328

Yu, B., Yin, H., & Zhu, Z. (2018). Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*. DOI: 10.24963/ijcai.2018/505

Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, *14*(1), 35–62. DOI: 10.1016/S0169-2070(97)00044-7

Zhang, J., Shi, X., Xie, J., Ma, H., King, I., & Yeung, D. Y. (2019). D-GCN: Dynamic graph convolutional networks for learning on point clouds. *arXiv preprint arXiv:1810.07836*.

Zhang, Y., Liu, S., Bai, Y., Ye, H., Yu, E., & Liu, Z. (2020). Spatio-temporal synchronous graph convolutional networks: A new framework for spatio-temporal network data forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(04), 6786–6793. DOI: 10.1609/aaai.v34i04.6158

Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., & Deng, M. (2019). T-GCN: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, *21*(9), 3848–3858. DOI: 10.1109/TITS.2019.2935152

Zhou, C., & Nelson, P. C. (2002). Predicting traffic congestion using recurrent neural networks. *9 th World Congress on Intelligent Transport Systems, Chicago, October 14-18, 2002, electronic proceedings.*

Prof. Pan Ruifeng has obtained a master's degree from Central South University for Nationalities. In over 20 years of professional experience, he has published multiple papers in journals/conferences, as well as 4 invention patents and several software copyrights. Currently, Prof. Pan Ruifeng serves as the Director of the Network Engineering Technology Teaching and Research Office at the School of Information Technology, Nanchang Vocational University. His research interests include network engineering technology, intelligent IoT applications, and artificial intelligence.

Mengsheng Wang is a lecturer at the Network Engineering Technology Teaching and Research Office, School of Information Technology, Nanchang Vocational University. His research focuses on computer network technology and network engineering education.

Jindan Zhang now is an associate professor in Xianyang Polytechnic Institute, she obtained her phd student in Xidian University, her main research interests include public key cryptography and cloud security. She has published about 15 papers in the field of information security including some top journals.

Prof Brij B. Gupta received the PhD degree from Indian Institute of Technology (IIT) Roorkee, India. In more than 17 years of his professional experience, he published over 500 papers in journals/conferences including 35 books and 10 Patents with over 21000 citations. He has received numerous national and international awards including Canadian Commonwealth Scholarship (2009), Best Faculty Award (2018 & 2019), etc. He is also selected in the 2021 and 2020 Stanford University's ranking of the world's top 2% scientists. He is also a visiting/adjunct professor with several universities worldwide. He is also an IEEE Senior Member (2017) and also selected as 2021 Distinguished Lecturer in IEEE CTSoc. Dr Gupta is also serving as Member-in-Large, Board of Governors, IEEE Consumer Technology Society (2022-2024). At present, Prof. Gupta is working as Director, International Center for AI and Cyber Security Research and Innovations, and Professor with the Department of Computer Science and Information Engineering (CSIE), Asia University, Taiwan. His research interests include cyber security, cloud computing, artificial intelligence, blockchain technologies, social media and networking.

Nadia Nedjah graduated in Systems Engineering and Computation (Ingeniorat en Informatique) - Université de Annaba/Algeria (09/1982 - 08/1987), Master of Science in Systems Engineering and Computation (Systèmes Informatiques) - Université de Annaba/Algeria (09/1987 - 08/1990)) and doctorate in Systems Engineering and Computation (Ph.D. in Computation) - University of Manchester - Institute of Science and Technology (04/1994 - 12/1997). She is currently a Full Professor at the State University of Rio de Janeiro. She has experience in Computer Science, with emphasis on Computer Systems, working mainly on topics related to Embedded Systems, Automatic Synthesis of Digital Circuits, FPGA Architectures, Distributed Systems, Encryption Hardware, Intelligent Systems and Artificial Intelligence Techniques, such as Swarm Intelligence, Artificial Neural Networks and Machine Learning, and Distributed Algorithms for Swarm Robotics. She is a member of the Board of Directors of the Brazilian Society of Computational Intelligence (SBIC) since 2015 - 2025. She was a member of the Board of Directors of the World Society of Engineering and Science during the period 2002 to 2006. She is Editor-in-Chief of the International Journal of Innovative Computing Applications (IJICA) and International Journal of High Performance Systems Architecture (IJHPSA), published by InderScience, Geneva/Switzerland, since 2007. She is also Associate Editor of various periódicos, such as Journal of Ambient Intelligence and Humanized Computing (Springer, IF: 7,104), International Journal of Bio-inspired Computation (InderScience, IF: 3,977), International Journal of Electronics (Taylor Francis, IF: 1,336), Microprocessors and Microsystems (Elsevier, IF: 1,525), Integration, The VLSI Journal (Elsevier, IF: 1,211) and IET Computers Digital Techniques (IET, IF: 0,803). From 10/2009 to 09/2013 and 12/2021 to 11/2023, she served as Coordinator of the research group in Intelligent Systems of the Postgraduate Program in Electronic Engineering at UERJ. Also, from 12/2021 to 04/2022, she served as Coordinator-Adjunct of the Postgraduate Program in Electronic Engineering at UERJ. She is the author of 3 books, more than 120 journal articles, 40 book chapters, and 180 conference articles. She organized 25 special issues for renowned journals, such as Applied Soft Computing (Elsevier, IF: 6,725), Neurocomputing (Elsevier, IF: 5,719) and Computers Electrical Engineering (Elsevier, IF: 3,818), and more than 48 edited books. So far, she is recipient of 5 UERJ's Prociência scholarships (2009 - 2024), 5 CNPq's Productivity scholarship (2007 - 2026) and 3 FAPERJ's Scientist of Our State (2015 - 2025). Currently, she is a permanent member of Le Conseil National Economique, Social et Environnemental of the Algerian Government.