A Novel Spatial-Temporal Multi-Scale Alignment Graph Neural Network Security Model for Vehicles Prediction

Chunyan Diao, Dafang Zhang, Wei Liang, Kuan-Ching Li, *Senior Member, IEEE*, Yujie Hong, and Jean-Luc Gaudiot, *Life Fellow, IEEE*

***Abstract*— Traffic flow forecasting is indispensable in today's society and regarded as a key problem for Intelligent Transportation Systems (ITS), as emergency delays in vehicles can cause serious traffic security accidents. However, the complex dynamic spatial-temporal dependency and correlation between different locations on the road make it a challenging task for security in transportation. To date, most existing forecasting frames make use of graph convolution to model the dynamic spatial-temporal correlation of vehicle transportation data, ignoring semantic similarity between nodes and thus, resulting in accuracy degradation. In addition, traffic data does not strictly follow periodicity and hard to be captured. To solve the aforementioned challenging issues, we propose in this article CRFAST-GCN, a multi-branch spatial-temporal attention graph convolution network. First, we capture the multi-scale (e.g., hour, day, and week) long- short-term dependencies through three identical branches, then introduce conditional random field (CRF) enhanced graph convolution network to capture the semantic similarity globally, so then we exploit the attention mechanism to captures the periodicity. For model evaluation using two real-world datasets, performance analysis shows that the proposed CRFAST-GCN successfully handles the complex spatial-temporal dynamics effectively and achieves improvement over the baselines at 50% (maximum), outperforming other advanced existing methods.**

***Index Terms*— Conditional random field, graph conventional network, smart city, security frame, vehicle prediction.**

Manuscript received September 16, 2021; revised November 11, 2021; accepted December 7, 2021. This work was supported by the National Natural Science Foundation of China (Grant No. 61976087 and Grant No. 62072170), the National Key Research and Development Program of China (No. 2021YFA1000600), the Fundamental Research Funds for the Central Universities (Grant No. 531118010527), the Key R & D Program of Hunan Province (Grant No. 2022GK2015), and the Hunan Provincial Natural Science Foundation of China (Grant No. 2021JJ30141). The Associate Editor for this article was A. Jolfaei. *(Corresponding author: Dafang Zhang.)*

Chunyan Diao, Dafang Zhang, and Yujie Hong are with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China (e-mail: cydiao@hnu.edu.cn; dfzhang59@163.com; hyj1998@hnu.edu.cn).

Wei Liang is with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China, and also with the Hunan Key Laboratory for Service Computing and Novel Software Technology, School of Computer Science and Engineering, Hunan University of Science and Technology, Xiangtan 411201, China (e-mail: weiliang99@hnu.edu.cn).

Kuan-Ching Li is with the Department of Computer Science and Information Engineering (CSIE), Providence University, Taichung 43301, Taiwan (e-mail: kuancli@pu.edu.tw).

Jean-Luc Gaudiot is with the Department of Electrical Engineering and Computer Science, University of California Irvine, Irvine, CA 92697 USA (e-mail: gaudiot@uci.edu).

Digital Object Identifier 10.1109/TITS.2022.3140229

I. Introduction

WITH the rapid development of cities and economical construction, the urbanized vehicle traffic system has also become increasingly more complex [1]. While the current traffic system brings convenience to people, it also leads to problems and issues incurred due to reduced visibility and massive greenhouse gas emissions [2]. For example, a complex road network makes the traffic system more congested and fragile. Moreover, cases of fatigue driving, illegal driving, and illegal occupation of emergency lanes have increasely occurred in recent years [3], [4], seriously jeopardizing the order and safety of vehicle driving and causing loss of life and property. Therefore, improving road traffic capacity, reducing road network maintenance costs, and accident risks are questions that need to be settled expeditiously by current intelligent traffic systems (ITS). In such systems, the short-term vehicle flow forecast problem is a critical yet challenging task. Advancements [5] in Big Data, smart cities, data storage, and urban computing make vehicle data collection and analysis possible, so the vehicle data (e.g., vehicle velocity, vehicle capacity, and pedestrian volume) reflect the current status of ITS and can be applied to predict the upcoming status for vehicle control.

The exponential growth of the IoT has provided an efficient approach to collect data for intelligent systems [6]. With improved distributed hardware scheduling algorithms [7], wireless sensor networks develop with low consumption and high fault tolerance [8],vehicle data can be collected through sensors that are deployed at a fixed position and sample regularly. However, observed values in the adjacent time slice are not independent despite dynamically correlated. In addition, vehicle flow data have apparent periodicity, two peaks corresponding to morning and evening. Intuitively, the traffic can impact the nearby road in a diffusion manner. Therefore, the road network vehicle data have a robust dynamic dependency, though it remains challenging to learn the internal spatial-temporal pattern and perform accurate vehicle prediction.

Vehicle flow forecasting is a classic spatial-temporal forecasting topic. Most works in the literature at early stage exploits time-series-based approaches to predict vehicle speed; for example, history average (HA) [9], vector autoregression

(VAR) [10], support vector regression (SVR) [11], Autoregressive Integrated Moving Average model (ARIMA) [12] and their variants. Unfortunately, the studies/methods developed have been assumed at a stationary basis, without spatial dependency. In addition, due to high uncertainty and complexity, they fail to forecast for long-term and extreme events (e.g., holidays, severe weather, and vehicle crash accidents). Therefore, with the impressive development of Deep Learning (DL), recent researches have been re-directed using this type of Artificial Intelligence (AI) technology for better accuracy and robustness [13].

DL-enabled approaches are widely applied in several spatial-temporal prediction tasks and outperform existing strategies. Recurrent Neural Networks (RNNs) are usually used to encode temporal patterns. Nevertheless, despite the tremendous success achieved by RNNs, the inadequate training and inference speed limits their further development, leading to an alternative solution for sequence modeling. It is known that can be benefited from Convolution Neural Networks (CNNs) of their effective parallelization, diffusely used in sequence modelings, speech recognition, natural language processing, and time series. However, CNNs can not handle data in non-euclidean space since the direct application of CNNs in time-series prediction may bring precision loss, unfortunately. The emergence of GCN provides the best choice for graph data learning. GCN can learn the characteristics and structural information of nodes simultaneously, and it is suitable for nodes of any topology. The results on the public data set show that the effect of GCN on tasks such as node classification and edge prediction is far superior to other solutions. Lastly the combination of GCN and time series methods also provides new ideas for traffic forecasting.

At this point, two key challenges are depicted in-vehicle prediction. Existing researches has ignored the shift of long cyclical dependence. Traffic data has a solid daily and weekly cyclical, which can be beneficial to prediction, and thus, the challenge is that vehicle data do not strictly follow the periodicity. For example, peak hours on weekdays always happen around 9:30am though they may vary from 3:00pm to 4:30pm on different days, while peak hours between weeks range from 8:30 to 9:30. Although previous works considered periodicity, but unfortunately, failed to capture the cyclical, as well as two sensors that have long-distance in spatial may share an implicit similarity. Therefore, due to the locality of the convolution operator, some useful information may be lost, so the preservation of semantic similarity remains challenging.

Based on the abovementioned issues, we propose in this article the CRFAST-GCN framework. Specifically, we develop a multi-branch architecture above trend, daily, weekly, and each branch contains a CFRGCN model and an attention component. The CRFGCN module captures the dynamic Spatial-temporal relationship while protecting the similarity information of the hidden layer of the graph convolution. The enhanced attention component to arrest the temporal pattern shift and align them, and next, a CRF-refined graph convolution layer is introduced to capture semantic similarity. In essence, our contributions in this article are threefolds:

- To develop a spatial-temporal multi-branch framework, named CRFAST-GCN, to learn dynamic and multi-scale correlation of traffic data. A temporal self-attention is applied to learn temporal pattern shift and alignment.
- To propose a CRF-based GCN block, which uses CRF between two graph convolution layers to preserve similarity and learn semantic similarity simultaneously.
- To conduct broad-scale experiments on two real-world vehicles datasets, the results show that CRFAST-GCN successfully handles the complex spatial-temporal dynamic, and its performance consistently outperforms baselines.

The remaining of this article is organized as follows. The related works of literature is presented in Section II, the notations and problems definition in Section III, and the proposed security frame in Section IV. Next, the experimental results, anslysis and discussions are depicted in Section V, and finally, the concluding remarks and prospect directions are drawn in Section VI.

II. Related Works

The Internet of Things (IoT) technology [14] continuously drives the evolution of vehicle networks into the Internet of Vehicles (IoV) paradigm, where a large amount of data acquired from connected vehicles, traffic monitoring, and flow augments the potential value of the Big Data in the IoV, improving traffic prediction toward intelligent transportation. In traffic prediction, prior knowledge is significant since grid-based data has an inherent limitation on representation complex spatial relationship [15], [16]. Therefore, predefined graph such as distance-based graph [17], binary graph [18], multi-graph [19], and self-adaptive graph [20], [21] are introduced. Also, many recent works adopts RNNs [22]–[25], Temporal Convolutional Networks (TCNs) [18], [20], [25], and transformer to model temporal dependency. Additionally, data are sensitive to external factors [26], so thus, existing related works in the field take weekday [27], points of interest [19], and weather condition [15], [28] into consideration for further improved accuracy.

A. Temporal Modeling

In the early times, traffic forecast was viewed as a naive sequentially prediction task. For instance, classic modus, AutoRegression Moving Average (ARMA) and ARIMA focus on univariate time series prediction without modeling dependent variables, limiting multi-variate time series forecast efficiency. Besides, these statistical approaches have a robust stationary assumption, leading to failure to predict unstable series and corrupted data. RNN is introduced to model sequential data due to its powerful ability to capture temporal dependency [24], [29]–[31]. Even though the recursive manner of RNNs brings flexibility in model design, the RNNs suffer from low computation efficiency and occupy massive memory. Moreover, it suffers from gradient vanishing/exploding, resulting in additional training difficulties. Hochreiter and Schmidhuber [32] designed a neural network called Long Short-Term Memory (LSTM) to solve gradient problems and enhance long-term memory. However, with a large number or longer-span sequences, as the depth of the network increases, the loss will be severe, also introducing the problem of overfitting. Meanwhile, Cho *et al.* [33] designed GRU to simplify LSTM. Recently, transformer [34] has made remarkable achievements in Natural Language Processing (NLP) and even sequentially forecasting. The self-attention mechanism [29], [35], [36] in the transformer can capture the global input-wise dependency with high parallelization potential though still suffer from large computation resources requirements. Generally speaking, time series-based forecasting methods ignore the temporal and spatial characteristics of traffic data. In fact, traffic flow has a strong spatial dependence. Congestion in a place will quickly affect the traffic of the surrounding road sections, which is also the challenge of considering only time series forecasting.

B. Spatial-Temporal Modeling

As known, GNNs have power and flexibility in capturing the spatial correlation in non-euclidean space data, so many works of the literature exploit them into time series prediction. For example, works presented in [17], [20]–[22] applied Graph Convolution Network (GCN [37]) and its variants in traffic prediction. Pan *et al.* [38] proposed meta-learning to generate parameters for Graph Attention Network (GAT) [39]. Li *et al.* [40] designed DCRNN to reformulate the spatial corrections of vehicles as an expanded procedure and separate the adjacent matrix into two directed graphs. Graph WaveNet [20], [29] combined GCN with a dilation causal convolution network to save the computational cost of processing long sequences and proposed adaptive adjacency matrix as a supplement to the predefined adjacency matrix to catch implicit spatial correlation. In recent work, Yu *et al.* [25] designed STGCN to capture spatial-temporal correlation through spatial graph convolution and temporal convolution. AGCRN can capture fine-grained temporal and spatial correlations of specific nodes in traffic sequences [41]. ASTGCN [18], STSGCN [39] and GMAN [35] further introduced a more carefully designed spatial-temporal attention mechanism to catch dynamic spatial-temporal correlation with GCN. Bai *et al.* [23] combined adaptive graph convolution GRU with an attention mechanism to capture temporal and spatial correlation dynamically. However, these methods only learned shared patterns of time series data. Although methods based on GCN capture the spatial correction and retrieve excellent results, graph data has similar information between different nodes, since preserving the similarity of the hidden layer in the graph convolution process cannot be ignored. Nevertheless, current research works do not aim to this direction. Although traffic flow data has prominent periodic characteristics, there is also cyclical volatility. Thereafter, the method proposed in this article solves these two abovementioned issues.

C. External Factors

The traffic conditions are vulnerable to external factors, such as weather [15], [29] and poi [40], so it is required to tackle it explicitly. Since the original time series can be used to generate time information [15], [17], [20], [40], [42], DTW similarity [43], the road structure and connectivity [19], [38] can be extracted from adjacency matrix. In addition, leveraging more external factors and designing effective ways to make factors cooperate well with each other is critical and promising. As a summary, the proposed model effectively deals with dynamic spatial similarity and period deviation through periodic attention mechanisms and conditional random fields, respectively.

III. Preliminary and Notation

A. Conditional Random Field

Conditional Random Field (CRF) was introduced in [44] for forecasting labels of sequence data and later extended its application to different structured prediction tasks [45], [46]. Essentially, CRF can forecast the future data of the node based on the information of the node and neighboring nodes. That is, maximize the probability of the predicted target, as:

$$P(y_t^i|x_t^i) = \frac{1}{Z(x_t^i)} \exp\left(-E(y_t^i|x_t^i)\right), \tag{1}$$

where $Z(x_t^i)$ is the normalization factor, $E(\cdot|\cdot)$ represents energy function, and y_t^i is a goal representation method. The energy function has a unary component and a binary component, where the former can predict the data of the node itself, and the latter can obtain the relevant data information between the node and its neighbor nodes to regularize the unary function. A node can predict its data for the next moment through its and neighbor nodes' information. For such, the defined energy function is:

$$E(y_t^i|x_t^i) = \Phi_u(y_t^i, x_t^i) + \sum_{j\in\mathcal{N}_i} \Phi_p(y_t^i, y_t^j, x_t^i, x_t^j) \tag{2}$$

where, $\Phi_u(y_t^i, x_t^i)$ is the unary energy function, and $\sum_{j\in\mathcal{N}_i} \Phi_p(y_t^i, y_t^j, x_t^i, x_t^j)$ is the pairwise function. Next, to adopt mean-field approximation to the majorization the CRF, based on the ability of CRF to capture the paired correlation between data points and their context. In this article, CRF and GCN are combined to maintain the similarity of the hidden layer.

B. Notations

In this investigation, we define the traffic network as an undirected graph $G = (V; E)$, where V represents the collection of nodes and E the connection edges of nodes. Table I depicts notations and descriptions in a task.

C. Data Segments

The traffic flow is classified into three segments, as trend, weekly, monthly, while definitions 1-3 define the meanings of different fields, in detail.

TABLE I
NOTATIONS AND DESCRIPTION IN CRFAST-GCN

Notation	Description
V	finite set of N nodes
E	connection edges of nodes
A	adjacent matrix of graph
x_t^i	all eigenvalues of node i at time t
X_t	all eigenvalues of all nodes at time t
$\mathcal{N}_i$	neighourhood node set of node i
$\mathcal{X}$	all eigenvalues fo all nodes over τ time slices

Definition 1: Trend segment:

$$\mathcal{X}_h = \left(\mathbf{X}_{t_0-T_h+1}, \mathbf{X}_{t_0-T_h+2}, \ldots, \mathbf{X}_{t_0}\right) \in \mathbb{R}^{N\times F\times T_h}, \quad (3)$$

Historical time series directly adjacent to the forecast period.

Definition 2: Weekly segment:

$$\mathcal{X}_d = [\mathbf{X}_{t_0-(T_d/T_p)*q+1}, \ldots, \mathbf{X}_{t_0-(T_d/T_p)*q+T_p}, \ldots, \mathbf{X}_{t_0-q+1}, \ldots, \mathbf{X}_{t_0-q+T_p}] \in \mathbb{R}^{N\times F\times T_d}, \quad (4)$$

Daily brunch is composed of segments of recent days in the same period as the forecast period.

Definition 3: Monthly segment:

$$\mathcal{X}_w = [\mathbf{X}_{t_0-7*(T_w/T_p)*q+1}, \ldots, \mathbf{X}_{t_0-7*(T_w/T_p)*q+T_p}, \ldots, \mathbf{X}_{t_0-7*q+1}, \ldots, \mathbf{X}_{t_0-7*q+T_p}] \in \mathbb{R}^{F\times N\times T_w}, \quad (5)$$

Segments of data from past few weeks matching with same week attributes and time intervals as forecast period.

IV. PROPOSED METHOD

A. Problem Definition

The problem is formulated by Equation (6), where $\mathcal{X}$ to predict the traffic Y of all nodes on the complete transportation network in the following time period.

$$Y = \mathcal{F}_{\Theta}(\mathcal{X}, A) \quad (6)$$

B. CRFAST-GCN Framework

From an intuitive point of view, the formation and spread of traffic congestion are gradual, so congestion at the previous moment directly impacts the next moment. Furthermore, traffic data may show repetitive patterns, such as morning rush hour during weekdays, due to people's daily routines. Therefore, accurate traffic flow prediction supports the road condition analysis and timely feedback traffic information to travelers. All three compositions use the same network architecture, consisting of several spatial-temporal blocks and a layer to connect them fully.

C. Spatial Graph Convolution

The two-dimensional convolution operation is developed to the graph structure with the application of the spectrogram theory. In this way, the transportation network can be represented as a graph, while a signal represents the nodes' characteristics. Shuman *et al.* proposed spectral CNN [47], where the diagonal matrix parameters are indexed as learnable parameters, and also assumed that the graph signal is multi-dimensional. The major disadvantage is that a sample's computational complexity is increased if it has a graph, given that it is required to perform the feature composition of the Laplace matrix U.

In spectrogram analysis, the Laplace matrix represents the graph, and the attributes of the graph structure are retrieved after detailed analysis. The Laplace matrix is $L = D - A$, where A is the adjacent matrix, and the diagonal matrix is $D \in R^{NxN}$, $D_{ii} = \sum_j A_{ij}$. The eigenvalue decomposition of the Laplace matrix is $L = U\Lambda U^T$. Taking the traffic flow signal at time t, the signal is $x = X_t^f \in R^N$, the graph signal $\hat{x} = U^t x$, and the corresponding Fourier inversion is $x = U\hat{x}$, given that the graph signal on the convolution operation is equal to the graphic Fourier transform product of transforming these signals into the spectral domain Simonovsky and Komodakis proposed [48]. Nevertheless, for large-scale graphs, it is expensive to decompose the eigenvalues of the Laplace matrix directly, so Chebyshev polynomials are selected in this article to approximate and effectively solve such a problem.

$$g_{\theta} *_G x = g_{\theta}(\mathbf{L})x = \sum_{k=0}^{K-1} \theta_k T_k(\widetilde{\mathbf{L}})x, \quad (7)$$

where $\theta \in R^K$ is a vector parameter of polynomial coefficients, $\tilde{L} = 2(L - I_N)/\lambda_{\max}$, where $\lambda_{\max}$ represents the maximum eigenvalue of L, and T_k is Chebyshev polynomial.

Chebyshev polynomial approximate expansion is adopted to solve the formula for extracting neighbors from 0 to K order around each node in the graph through the convolution kernel g. Simonovsky and Komodakis [49] proposed an effective variant of CNN that directly operates on the graph, and the local first-order approximation of spectrogram convolution can encourage the excellence of convolution structure. Thus, in the proposed multi-layer GCN, the stratification breeding rule is:

$$H^{(l)} = \mathrm{ReLU}\left(\hat{A}H^{(l-1)}W^{(1-1)}\right), \quad (8)$$

where ReLU is the active action and $W^{(l-1)}$ is the learned parameters in the $(l-1)$-th layer. $\hat{A}$ is defined as:

$$\hat{A} = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}} \quad (9)$$

D. Temporal Convolution

After capturing the neighboring information of the node in the spatial dimension through graph convolution operation, the information on the adjacent time slices of the signal of a node is updated by merging through a standard convolution layer in the time dimension. Thereafter, the operation of the first level in the time dimension is expressed as:

$$\mathcal{X}_h^{(l)} = \mathrm{Re}\,LU\left(\Phi * \left(\mathrm{ReL}\,U\left(g_{\theta *G}\widehat{\mathcal{X}}_h^{(l-1)}\right)\right)\right) \in \mathbb{R}^{C_r\times N\times T_r}, \quad (10)$$

where $*$ denotes a standard convolution operation, Φ is the temporal dimension convolution kernel, and ReLU as the activation function.

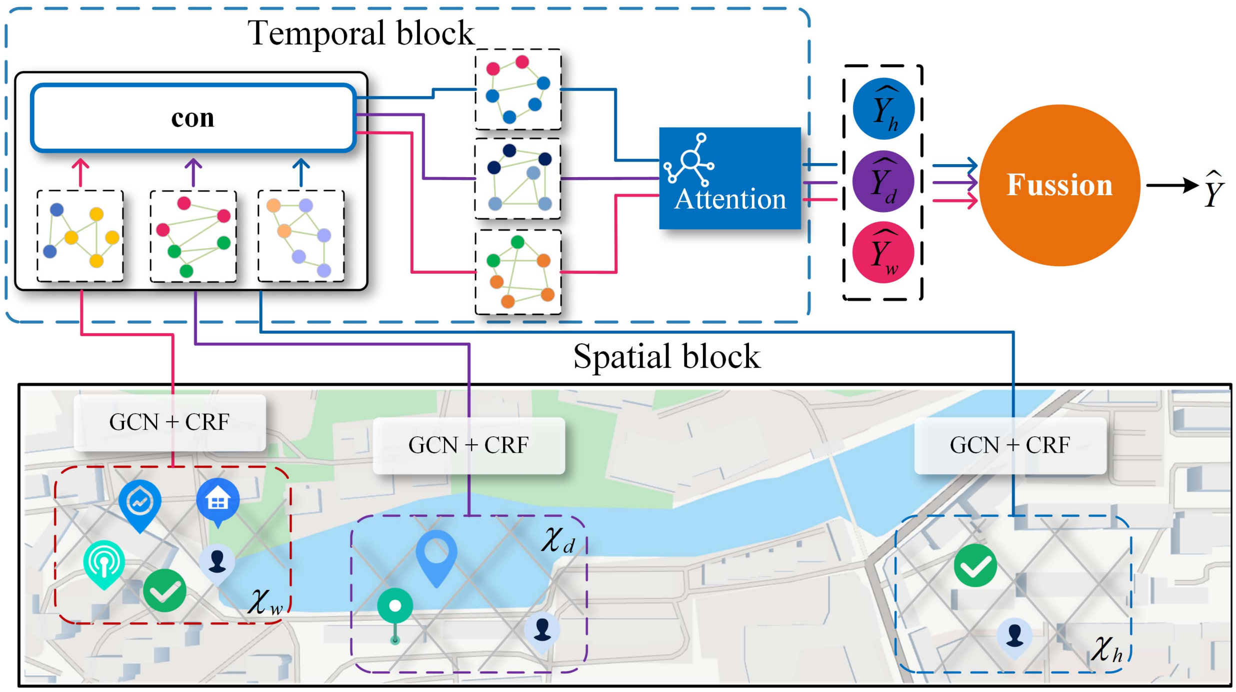


Fig. 1. CRFSTGCN frame,$\mathcal{X}_h$,$\mathcal{X}_d$,$\mathcal{X}_w$ captures the long and short cycle dependence, respectively. That is, GCN captures the spatial relationship, the CRF layer captures the graph convolution hidden layer similarity, con captures the time dimension of the long- and short-term periodic relationship, the attention mechanism captures the periodic volatility, and lastly, the fully connected layer outputs the prediction results.

In summary, the spatial-temporal convolution module can acquire data's temporal and spatial features skillfully since the spatial-temporal attention module and spatial-temporal convolution module consist of a spatial-temporal block. Furthermore, multiple spatial-temporal blocks are stacked further to extract a more extensive range of dynamic spatial-temporal correlations.

E. Spatial CRF-GCN Layer

The influence of traffic conditions between different nodes on spatial dimensions is highly dynamic. Compared with general data, the graph between other nodes hides parallel information, so it is challenging to maintain it in the GCN hidden layer. Next, a flexible CRF layer that can directly extract hidden similarity features between nodes in the GCN process is depicted in [50].

In this article, to maintain the similarity in the propagation of graph neural network, we borrow the idea from CRF that captures the pair-wise relation. Specifically, node representation of hidden layer $H^{(l)}$ can be regarded as random variable $\{H_i^{(l)}\}$, in which $H_i^{(l)}$ corresponding to node i. These random variable are under condition of $\{O_i^{(l)}\}$, where $O_i^l = \mathcal{F}_l(\hat{A}, H^{(l-1), W^{(l-1)}})$. Based on these facts, the CRF is formulated as:

$$P(H^{(l)}|O^{(l)}) = \frac{1}{Z(O^{(l)})}\exp(-E(H^{(l)}|O^{(l)})), \tag{11}$$

where $Z(\cdot)$ is normalize factor, and $E(\cdot)$ is energy function.

The energy function can be divided into the unary component and binary component. Herein, the unary function is defined as:

$$\Phi(H_i^{(l)}, O_i^{(l)}) = \left\| H_i^{(l)} - O_i^{(l)} \right\|_2^2. \tag{12}$$

To minimize this unary function, $O_i^{(l)}$ is forced to approximate to convolution output $H_i^{(l)}$. Next, to learn different similarity from hidden layer, pairwise energy function is applied, as:

$$\Phi_p(H_i^{(l)}, H_j^{(l)}, O_i^{(l)}, O_j^{(l)}) = g_{ij} \left\| H_i^{(l)} - H_j^{(l)} \right\|_2^2, \tag{13}$$

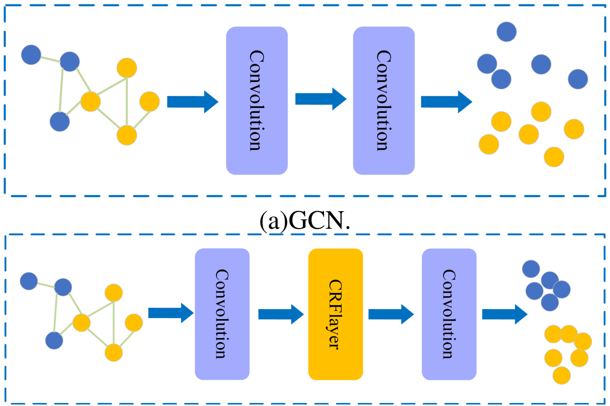
where g_{ij} is the similarity of nodes i and j. If g_{ij} is large, minimizing $\Phi_p(H_i^{(l)}, H_j^{(l)}, O_i^{(l)}, O_j^{(l)})$ will extort $H_i^{(l)}$ to get close to $H_j^{(l)}$. Otherwise, the distance between node i and j will be enlarged. subsequently, similar nodes are mapped to parallel position in hidden space. Finally, the energy function can be define as:

$$E(H_i^{(l)}|O_i^{(l)}) = \alpha \left\| H_i^{(l)} - O_i^{(l)} \right\|_2^2 + \beta \sum_{j \in \mathcal{N}_i} g_{ij} \left\| H_i^{(l)} - H_j^{(l)} \right\|_2^2, \tag{14}$$

Aimed to flexibly control the two components, two non-negative parameters α and β are introduced. The similarity fraction g_{ij} is computed by:

$$g_{ij} = \exp\left(\frac{\alpha O_j^{(l)} O_i^{(l)T}}{\left\| O_j^{(l)} \right\|_2 \left\| O_i^{(l)} \right\|_2} / \sigma^2 \right), \tag{15}$$

The approximation of $O_i^{(l)}$ and $H_i^{(l)}$ is strengthened by Equation 14, and compared to Equation 12, the pairwise components play a role in regularization.



(a)GCN.

(b)CRF layer of embedded GCN.

Fig. 2. Comparison of the graph CNN with the CRF layer, the designed CRF layer can better maintain the similarity of the output characteristics.

After defining Equation 14, we acquire a renew rule for the target representation $H^{(l)}$ through

$$\left(H_i^{(l)}\right)^k = \frac{\alpha O_i^{(l-1)} + \beta \sum_{j \in N_i} g_{ij} \left(H_i^{(l-1)}\right)^{k-1}}{\alpha + \beta \sum_{j \in N_i} g_{ij}}, \quad (16)$$

where N_i denotes the neighbor nodes set of node i, $\alpha > 0$ and $\beta > 0$ are used to process the balance of Equation 12, respectively.

To enable CRF to help comfort the similarity restrain in a graph convolution, a objective function is defined as:

$$\mathcal{L}_{CRF} = \sum_{l=1}^{L-1} \frac{\gamma}{2} \left\| H^{(l)} - F_l(\hat{A} H^{(l-1)}, W^{(l)}) \right\|_F^2, \quad (17)$$

where $\gamma > 0$, L is the number of layers of graph convolution network, and $\mathcal{L}$ is the objective function.

In mild condition, when γ increase, it is difficult to force H to satisfy similarity constraint. To solve this issue, the idea is to add regularization to $H^(l)$, and the objection function is depicted as:

$$\mathcal{L}_{CRF} = \sum_{l=1}^{L-1} \frac{\gamma}{2} \left\| H^{(l)} - F_l(\hat{A}, H^{(l-1)}, W^{(l)}) \right\|_F^2 + \mathcal{R}(H^{(l)}), \quad (18)$$

where $R(\cdot)$ is a regularization function to compute the back-propagation, and Fig. 2 shows the similarity of the output characteristics.

F. Attention for Time Cycle Shift

Once the graph convention similarity of the hidden layer is obtained, there is still a complex problem of time period fluctuations in the traffic flow. As shown in Fig. 3, the traffic data is not only periodic in long and short periods of time, as there is also cyclical volatility presented. For example, the peak period of yesterday morning was 9am, while today's peak period was 11am, and such a situation should bring errors to short-term predictions. To work out these challenges, this investigation proposes a cycle transfer mechanism based on attention.

At this point, we focus on solving daily cycle changes. Relative time interval P days include periodic processing dependencies. We further select the Q time interval for each day to solve the time shift problem. In Fig. 4, we select the attention mechanism to acquire the time shift and retrieve the weighted representation $\mathrm{x}_{i,t}^{p}$ of the previous day. As for the selected time interval, the total weight q defined is:

$$x_{i,t}^{p} = \sum_{q \in Q} \alpha_{i,t}^{p,q} x_{i,t}^{p,q}, \quad (19)$$

Comparing the previous hidden state $x_{i,t}^{p,q}$ with the temporal and spatial representations learned in short-term memory, we obtain $\alpha_{i,t}^{p,q}$, the weight formally defined as:

$$\alpha_{i,t}^{p,q} = \frac{\exp\left(\mathrm{score}\left(x_{\mathrm{i,t}}^{\mathrm{p,q}}, x_{\mathrm{i,t}}\right)\right)}{\sum_{q \in Q} \exp\left(\mathrm{score}\left(x_{\mathrm{i,t}}^{\mathrm{p,q}}, x_{\mathrm{i,t}}\right)\right)}, \quad (20)$$

Similar to [51], the score function defined in this work is given as:

$$\mathrm{score}\left(x_{\mathrm{i,t}}^{\mathrm{p,q}}, x_{\mathrm{i,t}}\right) = \mathbf{v}^{\mathrm{T}} \tanh\left(\mathbf{W}_{\mathrm{H}} x_{\mathrm{i,t}}^{\mathrm{p,q}} + \mathbf{W}_{\mathrm{X}} x_{\mathrm{i,t}} + \mathrm{b}_{\mathrm{X}}\right), \quad (21)$$

where $\mathbf{W_H}, \mathbf{W_X}, \mathbf{b_X}, \mathbf{v}$ are learned parameters, $\mathbf{v}^T$ denotes the transpose of $\mathbf{v}$.For each p of the previous day, we get the cycle representation. The time representation is:

$$\widehat{X}_{i,t}^{p} = \mathrm{ReLU}\left(X_{i,t}^{p}, \widehat{X}_{i,t}^{p-1}\right). \quad (22)$$

G. Multi-Scale Fusion

The way how to combine the output of these three components is depicted in this section. Take the traffic flow of the complete transportation network at 9:30 am as an example. In Fig. 3, it can be noted that the traffic flow of some weekdays has apparent peak hours in the morning, so the output of daily and weekly components may be more demanding and critical. However, there are no nitid traffic cycle patterns on other days, so daily and weekly cycle components may not help predict results. Thus, when the three components are fused, the output weight of each node is different, and it needs to learn from historical data. Conclusively, the final prediction result after fusion is:

$$\widehat{\mathbf{Y}} = \mathbf{W}_h \odot \widehat{\mathbf{Y}}_h + \mathbf{W}_d \odot \widehat{\mathbf{Y}}_d + \mathbf{W}_w \odot \widehat{\mathbf{Y}}_w, \quad (23)$$

where $\odot$ denotes inner product and $\mathbf{W}_h$, $\mathbf{W}_d$, and $\mathbf{W}_w$ are learning parameters, reflecting the influence degrees of the temporal-dimensional components on the forecasting target. Lastly, the prediction process of the entire framework is shown in Algorithm 1.

V. Experimental Results

The experimental processing on two real highway traffic data sets is carried out, and many comparative experiments to verify that the proposed model has the highest prediction accuracy are depicted.

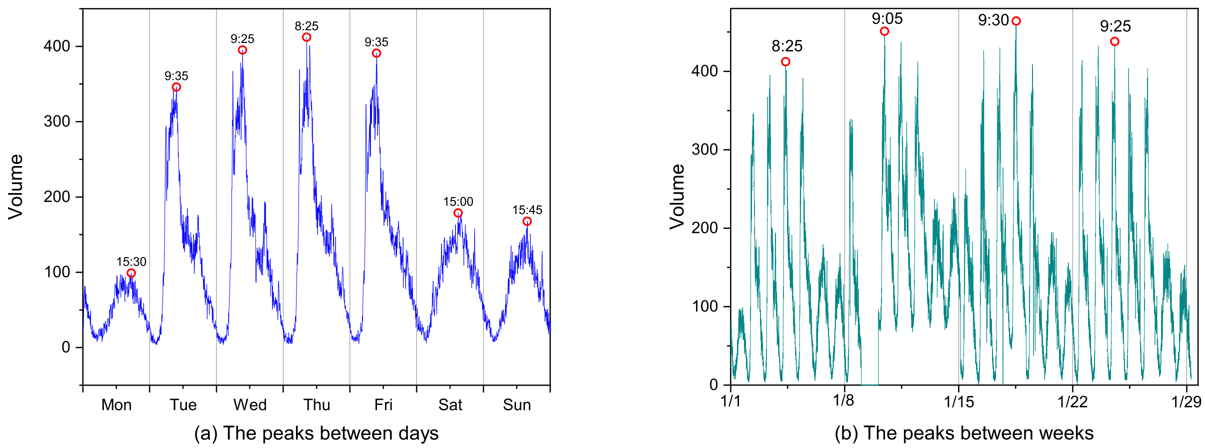


Fig. 3. Time offset of the cycle. (a) Time conversion between different days. (b) Time conversion between different weeks.

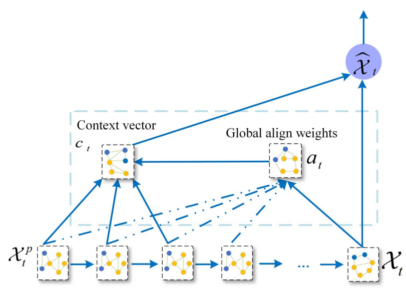


Fig. 4. In time step t, the weight vector a_t is aligned from the current target state $\mathcal{X}_t^p$ and all source states $\mathcal{X}_t$, and then calculates the global context vectorc_t according to a_t as the weighted average of all source states.

A. Evaluation Metrics

Three commonly used indicators are considered, the Root Mean Squared Errors (RMSE), Mean Absolute Errors (MAE), and Mean Absolute Percentage Error (MAPE), to evaluate the performance of all compared models as follows:

- **MAE**

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|x_i - \hat{x}_i\right| \tag{24}$$

- **RMSE**

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i - \hat{x}_i\right)^2} \tag{25}$$

- **MAPE**

$$\mathrm{MAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{\hat{x}_i - x_i}{x_i}\right| \tag{26}$$

where n is the number of test samples.

B. Experimental Datasets

All experiments in this article are conducted on PeMSD4 and PeMSD8. The Caltrans Performance Measurement System collects the datasets (PeMS) [52] in real-time at every 30 seconds, composed of 39000 detectors on the California highways, USA. The total flow, average occupancy, and average speed are considered as prediction indexes in this experiment.

Algorithm 1 Algorithm of CRFAST-GCN

Input: batch_size, num_of_vertices, num_for_prediction, all_backbones.
Output: Result.

if *len(all_backbones)* $>$ *0* **then**
  self.submodules = []
  with self.name_scope()
  for *backbones in all_backbones* **do**
    if *len(x_list) = len(self.submodules)* **then**
      if *len(num_of_vertices_set) = 1* **then**
        if *len(batch_size_set) = 1* **then**
          Result = nd.add_n(*submodule_outputs);
        end
      end
    end
  end
end
return Result;

TABLE II
STATISTICS OF PEMSD4 AND PEMSD8 DATASETS

Data	#Nodes	#Edges	#Timesteps	Time Ranges
PEMSD4	307	340	16992	1/1/2018-2/28/2018
PEMSD8	170	277	17856	1/7/2016-31/8/2016

The dataset is split into two parts: the first 50 days as the training set and the remaining data as the test set. Statistical properties and details of the abovementioned two datasets are shown in Table II.

C. Pre-Processing

This experiment utilizes linear interpolation to complete the missing value and zero-mean normalization data. The CRFAST-GCN model is implemented based on the MXNet framework, and several terms of the Chebyshev polynomial K are tested. As K increases, the prediction performance is slightly improved, and the same is true for the kernel size in the time dimension. Therefore, considering the improvement of computational efficiency and prediction performance, we

TABLE III

AVERAGE PERFORMANCE ON DIFFERENT METHODS OVER PEMSD4 AND PEMSD8

Model	PEMSD4			PEMSD8		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE
HA [9]	38.03	59.24	27.88	34.86	52.04	24.07
VAR [10]	24.44	37.76	17.27	19.21	29.74	13.09
LSTM [33]	25.72	39.67	17.71	20.40	31.55	12.77
GRU [33]	25.80	39.67	17.40	20.48	31.54	12.83
DCRNN [40]	21.08	32.74	14.62	16.10	25.00	10.28
STGCN [25]	22.89	35.39	14.94	19.00	28.83	11.78
ASTGCN [18]	22.69	34.65	16.23	18.97	28.39	12.70
STSGCN [39]	21.19	33.65	13.90	17.13	26.86	10.96
AGCRN [41]	19.75	32.25	13.05	16.00	25.21	10.55
CRFST-GCN(Ours)	20.82	31.13	14.94	15.24	24.23	10.19
CRFAST-GCN(Ours)	18.91	28.59	13.87	13.46	22.70	9.88

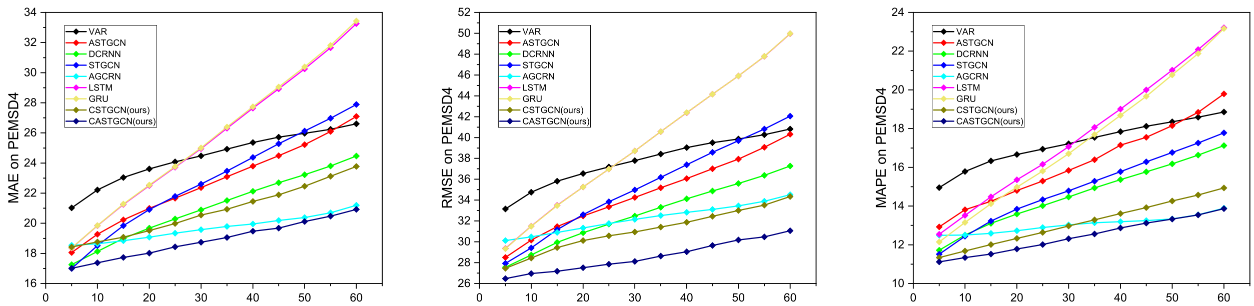


Fig. 5. Performance evaluation with increasing prediction interval on PEMSD4.

set K = 3, and the kernel size along the time dimension is 3 [18].

In this model, all graph convolution and temporal convolution layers use 64 convolution kernels. The data is adjusted by controlling the step size of time-domain convolution. In this research, we aim to predict the traffic flow for the next coming hour. The MSE between prediction and label is used as the objective function, and the error is minimized by backpropagation. During training, the batch size is 64, and learning rate is 0.0001. [18]

D. Performance Evaluation

The proposed model is compared with nine benchmark methods on PeMSD4 and PeMSD8, and among all results, the proposed CRFAST-GCN has achieved the best result among all evaluation indicators, as shown in Table III the predicted data results for the next hour.

As seen from the Table, the accuracy of traditional sequential analysis methods is not high enough since they ignore the spatial correlation. Specifically, HA performance is not favorable in traditional prediction methods based on time series since they only rely on the historical record of the predicted value and ignore the spatial and other contextual features. Despite the remaining baselines achieving better results than traditional sequential methods, they could not acquire the complicated nonlinear temporal features and dynamic spatial relationships. Therefore, the proposed method is significantly better than those based on regression. Moreover, LSTM models ignore the dynamic spatial similarity and periodic time transition.

The proposed model's excellent performance demonstrates the availability of the periodic systematic diversion mechanism in capturing the dynamic temporal and spatial similarity. Among them, models considering both time and space correlation, including STGCN, ASTGCN, STSTGCN, DCRNN, and two models proposed are superior to traditional DL techniques (LSTM, GRU). DCRNN combines graphics convolution and diffusion processes and combines GCN and GRU in an encoder mode to perform multi-step prediction; however, loss in this iterative process may cause errors. AGCRN utilizes multi-layer superposition as an encoder to capture nodes' temporal and spatial correlation and applies a linear transformation to predict the next time step. However, this method does not consider the preservation of the hidden layer similarity during the convolution process. In addition, the performance of ASTGCN is remarkable than STGCN, indicating that the energetic changes of data flow can be controlled by multi-level attention. Besides, to verify the influence of the periodic fluctuation attention mechanism proposed in this research, we designed a degraded version of CRFAST-GCN named CRFST-GCN, which outperforms most state-of-the-art baselines without any attention mechanism with nitid advantages when describing the peculiarity of vehicle data.

The results over different baselines are depicted in Figure 5, where we observe that with the enlarge of prediction length, the forecast error increases. Additionally, in short interval forecasting, methods as LSTM and GRU only use time correlation as an indicator and can obtain higher prediction accuracy. Nevertheless, with the increases in forecasting length, the prediction accuracy drops sharply. In contrast, performance

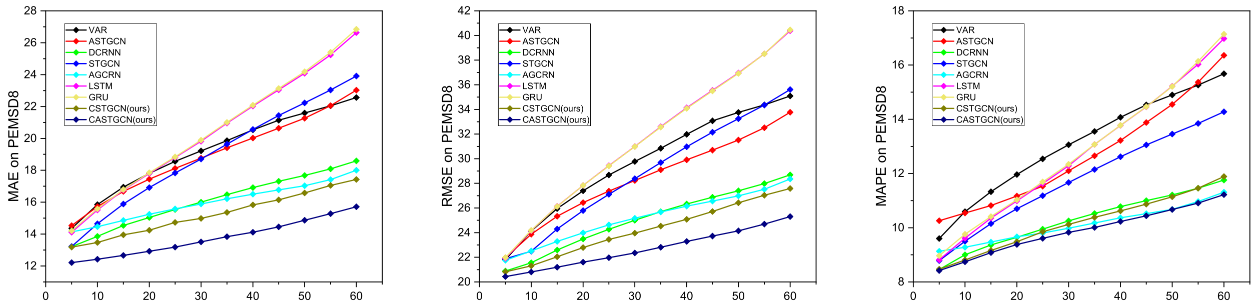


Fig. 6. Performance evaluation with increasing prediction interval on PEMSD8.

TABLE IV
COMPUTATION COST ON THE PEMSD4 DATASET

Model	#Parameters	#Training Time(epoch)
DCRNN	149057	36.39 s
STGCN	211596	16.36 s
ASTGCN	450031	49.47 s
CRFAST-GCN	886272	35.14 s

VAR declines more slowly than these methods in Figure 6, mainly because VAR can also consider the more critical spatial-temporal correlation in long-term forecasting. However, when the scale of the transportation network becomes more prominent, when more time series are considered in the model, the prediction error of VAR increases. As a result, its performance on PeMSD4 is worse than that on PeMSD8. The error of the deep learning method increases slowly with the enlargement of the forecasting length. In short, the overall completion is better.

The attention mechanism of the proposed method is significantly different from GAT, as it occurs before the activation function, while the proposed one occurs oppositely - after the activation. Nonlinear functions cannot guarantee the preservation of the structure of the node distribution, and the semantic similarity may lose after the convolution operation, so thus, the proposed CRFAST-GCN model consistently achieves the best predictive performance. The gap between CRFAST-GCN and other approaches is more nitid in long-term forecasting, indicating that combining attention mechanism and conditional random field graph convolution can better mine traffic data's dynamic spatial-temporal patterns and periodicity.

E. Computation Cost

Computation cost is calculated by comparing the parameter numbers and training time on the PeMSD4 dataset. It is shown in Table IV that the CRFAST-GCN parameter is six times of the DCRNN, which coresponds to the price of solving the periodic shift. The performance of AGCRN is slightly faster than DCRNN since the time convolution structure is used, similar to STGCN. However, ASTGCN requires more training time to improve the spatial and temple attention mechanisms to learn more accurate spatial-temporal patterns. Finally, in terms of significant performance improvements in Table III, AGCRN has a satisfactory computational cost.

VI. CONCLUSION AND FUTURE WORK

With advances in computation and communication technologies, traffic flow prediction is highly important for intelligent traffic management security, applying several technologies and strategies. We propose in this article an effective spatial-temporal multi-scale alignment graph neural network security model, which is based on the conditional random field and able to effectively extract the similarity of the graph convolution hidden layer and capture the dynamic spatial similarity of traffic data. The attention mechanism solves the problem of cycle offset by extracting data for one hour before and after the predicted time period and effectively aligning the data. Experimental results show that the proposed model outperforms other related models in the field. In fact, traffic data is sensibly influenced by various factors such as cycle shifts, weather, temperature, seasonal variation, spatial relationships, functions of road, cross-sections, and others. As future work, we will consider the influence of various factors when designing the model to improve the prediction accuracy further. Since CRFAST-GCN is a general Spatial-temporal data prediction model, we can also apply it to other applications, such as predicting speed, estimating the time required to reach the destination, etc.

REFERENCES

[1] Z. Xu, W. Liang, K.-C. Li, J. Xu, and H. Jin, "A blockchain-based roadside unit-assisted authentication and key agreement protocol for Internet of Vehicles," *J. Parallel Distrib. Comput.*, vol. 149, pp. 29–39, Mar. 2021.

[2] H. Su, M. Qiu, and H. Wang, "Secure wireless communication system for smart grid with rechargeable electric vehicles," *IEEE Commun. Mag.*, vol. 50, no. 2, pp. 62–68, Aug. 2012.

[3] L. Nkenyereye, B. Adhi Tama, M. K. Shahzad, and Y.-H. Choi, "Secure and blockchain-based emergency driven message protocol for 5G enabled vehicular edge computing," *Sensors*, vol. 20, no. 1, p. 154, Dec. 2019.

[4] Z. Feng, J. Zhan, J. Liu, W. Zhang, M. Zhou, and K. Wang, "Development and validation of highway emergency lane illegal occupancy scale for Chinese drivers," *Transp. Res. Rec.*, vol. 2634, no. 1, pp. 43–49, Jan. 2017.

[5] W. Liang, Y. Fan, K.-C. Li, D. Zhang, and J.-L. Gaudiot, "Secure data storage and recovery in industrial blockchain network environments," *IEEE Trans. Ind. Informat.*, vol. 16, no. 10, pp. 6543–6552, Oct. 2020.

[6] W. Liang, S. Xie, J. Long, K.-C. Li, D. Zhang, and K. Li, "A double puf-based rfid identity authentication protocol in service-centric Internet of Things environments," *Inf. Sci.*, vol. 503, pp. 129–147, Dec. 2019.

[7] X. Tang, K. Li, M. Qiu, and E. H.-M. Sha, "A hierarchical reliability-driven scheduling algorithm in grid systems," *J. Parallel Distrib. Comput.*, vol. 72, no. 4, pp. 525–535, Apr. 2012.

[8] M. Qiu, Z. Ming, J. Li, J. Liu, G. Quan, and Y. Zhu, "Informer homed routing fault tolerance mechanism for wireless sensor networks," *J. Syst. Archit.*, vol. 59, nos. 4–5, pp. 260–270, Apr./May 2013.

[9] D. Garrow, "Odd deposits and average practice. A critical history of the concept of structured deposition," *Archaeol. Dialogues*, vol. 19, no. 2, pp. 85–115, Dec. 2012.

[10] E. Zivot and J. Wang, "Vector autoregressive models for multivariate time series," in *Proc. Modeling Financial Time*, 2006, pp. 385–429.

[11] H. Drucker, C. J. Burges, L. Kaufman, A. J. Smola, and V. Vapnik, "Support vector regression machines," in *Proc. Neural Inf. Process. Syst.*, vol. 9, 1997, pp. 155–161.

[12] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results," *J. Transp. Eng.*, vol. 129, no. 6, pp. 664–672, Nov. 2003.

[13] W. Liang, L. Xiao, K. Zhang, M. Tang, D. He, and K.-C. Li, "Data fusion approach for collaborative anomaly intrusion detection in blockchain-based systems," *IEEE Internet Things J.*, early access, Jan. 22, 2021, doi: 10.1109/JIOT.2021.3053842.

[14] W. Liang, Z. Ning, S. Xie, Y. Hu, S. Lu, and D. Zhang, "Secure fusion approach for the Internet of Things in smart autonomous multi-robot systems," *Inf. Sci.*, vol. 579, pp. 468–482, Jun. 2021.

[15] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. 21st AAAI Conf. Artif. Intell.*, 2017, pp. 1655–1661.

[16] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li, "Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction," in *Proc. Conf. Artif. Intell.*, vol. 33, 2019, pp. 5668–5675.

[17] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," 2017, *arXiv:1707.01926*.

[18] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *Proc. AAAI*, vol. 33, 2019, pp. 922–929.

[19] G. Xu, Y. Li, L. Wang, L. Zhang, Q. Yang, J. Ye, and Y. Liu, "Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, pp. 3656–3663.

[20] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph WaveNet for deep spatial-temporal graph modeling," 2019, *arXiv:1906.00121*.

[21] Q. Zhang, J. Chang, G. Meng, S. Xiang, and C. Pan, "Spatio-temporal graph structure learning for traffic forecasting," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2020, vol. 34, no. 1, pp. 1177–1185.

[22] W. Chen, L. Chen, Y. Xie, W. Cao, and X. Feng, "Multi-range attentive bicomponent graph convolutional network for traffic forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 4, pp. 3529–3536.

[23] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, "Adaptive graph convolutional recurrent network for traffic forecasting," 2020, *arXiv:2007.02842*.

[24] Z. Cui, K. Henrickson, R. Ke, and Y. Wang, "Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, pp. 4883–4894, Nov. 2020.

[25] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," 2017, *arXiv:1709.04875*.

[26] W. Liang *et al.*, "Deep neural network security collaborative filtering scheme for service recommendation in intelligent cyber-physical systems," *IEEE Internet Things J.*, early access, Jun. 7, 2021, doi: 10.1109/JIOT.2021.3086845.

[27] K. Guo, Y. Hu, Z. Qian, Y. Sun, J. Gao, and B. Yin, "Dynamic graph convolution network for traffic forecasting based on latent network of Laplace matrix estimation," *IEEE Trans. Intell. Transp. Syst.*, early access, Sep. 9, 2020, doi: 10.1109/TITS.2020.3019497.

[28] W. Liang, J. Long, K.-C. Li, J. Xu, N. Ma, and X. Lei, "A fast defogging image recognition algorithm based on bilateral hybrid filtering," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 17, no. 2, Apr. 2021, doi: 10.1145/3391297.

[29] X. Wang *et al.*, "Traffic flow prediction via spatial temporal graph neural network," in *Proc. Web Conf.*, 2020, pp. 1082–1092.

[30] Z. Zhang, M. Li, X. Lin, Y. Wang, and F. He, "Multistep speed prediction on traffic networks: A deep learning approach considering spatio-temporal dependencies," *Transp. Res. C, Emerg. Technol.*, vol. 105, pp. 297–322, Oct. 2019.

[31] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, and G. Cottrell, "A dual-stage attention-based recurrent neural network for time series prediction," 2017, *arXiv:1704.02971*.

[32] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[33] K. Cho *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, *arXiv:1406.1078*.

[34] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[35] C. Zheng, X. Fan, C. Wang, and J. Qi, "GMAN: A graph multi-attention network for traffic prediction," in *Proc. AAAI*, 2020, vol. 34, no. 1, pp. 1234–1241.

[36] M. Xu *et al.*, "Spatial-temporal transformer networks for traffic flow forecasting," 2020, *arXiv:2001.02908*.

[37] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.

[38] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, and J. Zhang, "Urban traffic prediction from spatio-temporal data using deep meta learning," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2019, pp. 1720–1730.

[39] P. V. kovià, G. Cucurull, A. Casanova, A. Romero, P. Liü, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.

[40] Z. Lin, J. Feng, Z. Lu, Y. Li, and D. Jin, "DeepSTN+: Context-aware spatial-temporal neural network for crowd flow prediction in metropolis," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, Jul. 2019, pp. 1020–1027.

[41] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, "Adaptive graph convolutional recurrent network for traffic forecasting," 2020, *arXiv:2007.02842*.

[42] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, "Connecting the dots: Multivariate time series forecasting with graph neural networks," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2020, pp. 753–763.

[43] H. Yao *et al.*, "Deep multi-view spatial-temporal network for taxi demand prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2018, vol. 32, no. 1, pp. 2588–2595.

[44] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," Jun. 2001.

[45] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2017.

[46] D. Chen, D. Xu, H. Li, N. Sebe, and X. Wang, "Group consistent similarity learning via deep CRF for person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8649–8658.

[47] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2012.

[48] M. Henaff, J. Bruna, and Y. LeCun, "Deep convolutional networks on graph-structured data," 2015, *arXiv:1506.05163*.

[49] M. Simonovsky and N. Komodakis, "Dynamic edge-conditioned filters in convolutional neural networks on graphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3693–3702.

[50] H. Gao, J. Pei, and H. Huang, "Conditional random field enhanced graph convolutional neural networks," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2019, pp. 276–284.

[51] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," 2015, *arXiv:1508.04025*.

[52] C. Chen, K. Petty, A. Skabardonis, P. Varaiya, and Z. Jia, "Freeway performance measurement system: Mining loop detector data," *J. Transp. Res. Board*, vol. 1748, no. 1, pp. 96–102, Jan. 2001.

Chunyan Diao received the bachelor's and master's degrees from the School of Computer Science and Technology, Henan Normal University, in 2013 and 2017, respectively. She is currently pursuing the Ph.D. degree with the College of Computer Science and Electronic Engineering, Hunan University. Her research interests include authentication of graph data and vehicle security prediction.

Dafang Zhang received the Ph.D. degree in applied mathematics from Hunan University, Changsha, China, in 1997. He is currently a Professor with the College of Computer Science and Electronic Engineering, Hunan University. He was a Visiting Fellow with Regina University, Regina, SK, Canada, from 2002 to 2003, and a Senior Visiting Fellow with Michigan State University, East Lansing, MI, USA, in 2013. He has authored or coauthored more than 230 journals/conference papers and the principal investigator (PI) for more than 30 large-scale scientific projects. His research interests include dependable systems/networks, network security, network measurement, hardware security, and IP protection.

Wei Liang received the Ph.D. degree in computer science and technology from Hunan University, China, in 2013. He was a Post-Doctoral Scholar with Lehigh University, Bethlehem, PA, USA, from 2014 to 2016. He is currently a Professor with the School of Computer Science and Engineering, Hunan University of Science and Technology. He has authored or coauthored more than 140 journals/conference papers, such as IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTAION SYSTEMS, IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTING, IEEE TRANSACTIONS ON COMPUTATIONAL BIOLOGY AND BIOINFORMATICS, and IEEE INTERNET OF THINGS JOURNAL. His research interests include blockchain security technology, networks security protection, embedded systems and hardware IP protection, fog computing, and security management in wireless sensor networks (WSN).

Kuan-Ching Li (Senior Member, IEEE) received the Licenciatura degree in mathematics and the M.S. and Ph.D. degrees in electrical engineering from the University of Sao Paulo (USP), Brazil, in 1994, 1996, and 2001, respectively. He is currently a University Distinguished Professor with the Department of Computer Science and Information Engineering (CSIE), Providence University, Taiwan, where he also serves as the Director for the High-Performance Computing and Networking Center. He published more than 350 scientific papers and articles and is the coauthor or the co-editor of more than 30 books published by Taylor and Francis, Springer, and McGraw-Hill. His research interests include parallel and distributed computing, big data, and emerging technologies. He is the editor-in-chief of connection science and also serves at leading positions for several scientific journals. Also, he has been actively involved in many major conferences and workshops in program/general/steering conference chairperson positions and has organized numerous conferences and workshops.

Yujie Hong received the bachelor's degree from the School of Computer Science and Engineering, Hunan University of Science and Technology, in 2020. He is currently pursuing the Ph.D. degree with the College of Computer Science and Electronic Engineering, Hunan University, China. His research interests include blockchain and distributed systems.

Jean-Luc Gaudiot (Life Fellow, IEEE) received the Diplôme d'Ingénieur degree from the École Supérieure d'Ingénieurs en Electronique et Electrotechnique, Paris, France, in 1976, and the M.S. and Ph.D. degrees in computer science from the University of California, Los Angeles, in 1977 and 1982, respectively. He is currently a Distinguished Professor with the Department of Electrical Engineering and Computer Science, UC Irvine. Prior to joining UCI in 2002, he has been a Professor of electrical engineering with the University of Southern California since 1982. He has authored or coauthored over 250 journals and conference papers. His research has been sponsored by NSF, DoE, DARPA, and a number of industrial companies. His research interests include multithreaded architectures, fault-tolerant multiprocessors, and implementation of reconfigurable architectures. He has served the community in various positions. He was the President of the IEEE Computer Society in 2017.