A GAN-Based Short-Term Link Traffic Prediction Approach for Urban Road Networks Under a Parallel Learning Framework

Junchen Jin, *Member, IEEE*, Dingding Rong, Tong Zhang, Qingyuan Ji, Haifeng Guo, *Member, IEEE*, Yisheng Lv, *Senior Member, IEEE*, Xiaoliang Ma, and Fei-Yue Wang, *Fellow, IEEE*

***Abstract*— Road link speed is often employed as an essential measure of traffic state in the operation of an urban traffic network. Not only real-time traffic demand but also signal timings and other local planning factors are major influential factors. This paper proposes a short-term traffic speed prediction approach, called PL-WGAN, for urban road networks, which is considered an important part of a novel parallel learning framework for traffic control and operation. The proposed method applies Wasserstein Generative Adversarial Nets (WGAN) for robust data-driven traffic modeling using a combination of generative neural network and discriminative neural network. The generative neural network models the road link features of the adjacent intersections and the control parameters of intersections using a hybrid graph block. In addition, the spatial-temporal relations are captured by stacking a graph convolutional network (GCN), a recurrent neural network (RNN), and an attention mechanism. A comprehensive computational experiment was carried out including comparing model prediction and computational performances with several state-of-the-art deep learning models. The proposed approach has been implemented and applied for predicting short-term link traffic speed in a large-scale urban road network in Hangzhou, China. The results suggest that it provides a scalable and effective traffic prediction solution for urban road networks.**

***Index Terms*— Short-term link speed prediction, signalized urban networks, Wasserstein generative adversarial network.**

Manuscript received 25 March 2020; revised 19 August 2021 and 15 November 2021; accepted 19 January 2022. Date of publication 14 February 2022; date of current version 12 September 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2020YFB2104001, in part by the National Natural Science Foundation of China under Grant U1811463 and Grant 52072343, in part by the Zhejiang Provincial Natural Science Foundation (ZJNSF) under Grant LY20E080023, in part by the iTensor Project by Richterska Stiftelsen under Grant 2019-00498, and in part by the iHorse Project by KTH Digital Futures. The Associate Editor for this article was Y. Gao. *(Corresponding author: Qingyuan Ji.)*

Junchen Jin is with the College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China.

Dingding Rong is with Enjoyor Company Ltd., Hangzhou 310030, China.

Tong Zhang is with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China.

Qingyuan Ji is with Enjoyor Company Ltd., Hangzhou 310030, China, and also with the College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China (e-mail: qingyuan.ji@zju.edu.cn).

Haifeng Guo is with the College of Information Engineering, Zhejiang University of Technology, Hangzhou 310013, China.

Yisheng Lv and Fei-Yue Wang are with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China.

Xiaoliang Ma is with the Department of Civil and Architecture Engineering, School of Architecture and Building Environment and Centre of Digital Futures, School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, 10044 Stockholm, Sweden.

Digital Object Identifier 10.1109/TITS.2022.3148358

I. Introduction

THE explosive increase of traffic data, especially floating car data (FCD), has promoted the development of intelligent transport system (ITS) technologies during the past decade. Nowadays, FCD data has been broadly used for monitoring and operations of road traffic on large networks. In an urban environment, network traffic states are often represented by link travel time or traffic speed. Moreover, link traffic speed is considered an important measure for urban traffic control and management in engineering practice [1]. Traffic states of a road link normally have some spatial and temporal correlations with its neighboring links. At an intersection, the associated road links have complicated spatial-temporal impacts on each other by the pattern of traffic turning movements. At adjacent intersections, the influence of a road link on the traffic speeds of other associated links varies over time. Traffic flows on an urban network are often governed by signal controls. Link traffic speed is therefore influenced not only by traffic conditions of associated links but also by traffic control parameters and other planning factors at the intersection.

In this study, a deep learning (DL) based approach is proposed for predicting link traffic speed of urban road network, and the method takes advantage of a Wasserstein Generative Adversarial Nets (WGAN) in formulating the structure of DL model [2]. Two deep neural networks (DNN) are deployed as the generator and discriminator respectively and carry out the machine learning process using an adversarial mechanism. The main idea is that the WGAN-based model may achieve a more robust prediction performance. In addition, graph convolutional network (GCN) [3], recurrent neural network (RNN) [4] are also adopted in our study to model the spatial and temporal correlations of link traffic states.

We complement the previous studies [5], [6], and address data-driven traffic modeling under a comprehensive machine learning framework, called parallel learning (PL) framework [7], for advanced urban traffic control and management. The conceptual framework of PL is briefly described by

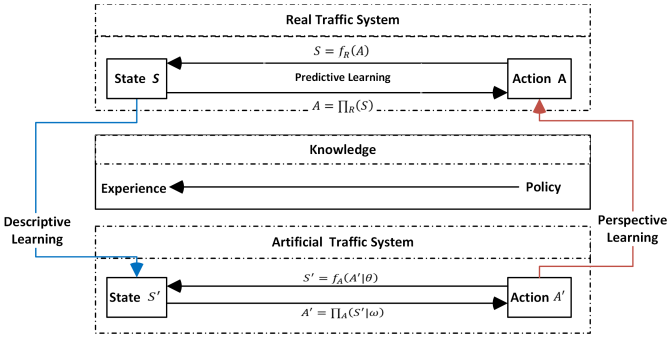


Fig. 1. The conceptual framework of the PL approach [7].

Figure 1. The PL framework synchronizes a real-world traffic system with an artificial traffic system [8]. Three learning processes are formulated within the framework: descriptive learning, predictive learning, and prescriptive learning. Traffic states, the derived knowledge, and control actions form an abstract loop. The framework aims to iteratively improve the operational execution of the real traffic system by self-boosting processes. The loop starts from a descriptive learning process forming a self-consistent system that coincides with observations from the real-world [5], [6]. To guide the real traffic system, the loop ends with a prescriptive learning process where optimal control policy is determined in the artificial simulation system that can be simultaneously adopted in the real system [9]. The proposed approach focuses on the predictive learning process of PL to connect descriptive learning to prescriptive learning. The relationships between traffic state (i.e., link speed) and traffic control (i.e., signal timing) are explored in our study.

The content of this paper is organized as follows. The recent developments of short-term traffic prediction for urban road networks are summarized in Section II. Section III then illustrates the essential components of the proposed data-driven prediction approach on link traffic state. A case study was carried out to verify the present approach using real-world data collected in the city of Hangzhou, China. Section IV presents the corresponding evaluation details and results. Pending research challenges are discussed in section V while the guidelines for future research are also provided.

II. Literature Review

Three challenges have been witnessed for traffic control and management in modern ITS systems: prediction accuracy, decision-making latency, and partial coverage of the managed area. In terms of large-scale real-time traffic management or information provision problems, dynamic traffic assignment (DTA) is a viable method that implements route assignment using an iterative procedure. The prediction of drivers' response behavior is important for traffic management strategies, such as route guidance [10]. For instance, a fuzzy control modeling approach is used to determine the behavior-consistent routing strategies in real-time [11]. The prediction accuracy of traffic states is however of the utmost importance for such proactive area-wide traffic management measures.

Traditionally, short-term traffic prediction primarily relies on mathematical and statistical modeling approaches, such as the autoregressive moving average model [12], and Kalman filter [13], [14]. With the abundance of traffic monitoring data collected from intelligent transportation systems, data-driven approaches are deemed as a prominent solution for traffic state prediction problems, e.g., [15]. Among the modeling efforts, various DNNs have been applied to forecast short-term traffic flow, speed, and travel time, such as stacked autoencoders [16], deep belief network [17], convolutional neural networks (CNNs) [18], and long short-term memory neural network (LSTM) [19]. Historical traffic data was the primary information source in these studies, from which dynamic traffic states used in prediction were derived for a single facility (e.g., motorway or arterial road). Although spatial-temporal approaches were introduced for highway traffic prediction e.g., [20], few studies have explored those correlations in urban traffic networks. Especially, traffic states in an urban network are influenced by not only surrounding traffic demand but also traffic signal controls at intersections. Few studies have taken into account traffic control parameters (i.e. signal timings for urban road intersections) for short-term traffic prediction.

To represent spatial correlations in a road network, graph convolutional network (GCN) has become an increasingly popular deep learning tool. For example, GCNs have been used in short-term traffic prediction due to their advantages in learning spatial-temporal dependencies over arbitrary graphs [21], [22]. GAN provides a path to data augmentation in many domains, including traffic prediction [23]. An important extension to GAN is in their use of conditionally generating an output by a deep conditional generative adversarial network (DCGAN) approach [24], where spatial and temporal correlations are conditionally captured in such a process. The DCGAN-based model may suffer from some issues of the unstable training process, whereas WGAN (Wasserstein GAN) is a recent feasible solution for this problem [2].

III. Methodology

A. Problem Definition

This study aims to establish a DNN model F_{dnn} that enables short-term link traffic prediction on a large road network. The traffic speed of a road link, associated with a signalized intersection, is forecasted using past observations. The prediction horizon based on the average length of signal cycles is marked as T_c. The past observations include both link traffic speed and traffic signal data of the associated intersection and its neighboring intersections.

Let $\mathbf{x}_t$, $\boldsymbol{\sigma}_t$, and $\mathbf{y}_t$ represent the vector of traffic inputs, signal timing, and the vector of predicted link traffic speed associated with current time t, respectively. Assume the model predicts link traffic speeds for a forecasting period of $\theta \cdot T_c$ time steps. Whereas, the data used for prediction in the model is from a period of previous $\alpha \cdot T_c$ time steps. The problem can be analytically represented by

$$\begin{aligned}\mathbf{Y} &= [\mathbf{y}_{t+1}, \ldots, \mathbf{y}_{t+p}, \ldots, \mathbf{y}_{t+\theta \cdot T_c}]^T \\ &= F_{dnn}\big(\mathbf{x}_{t-\alpha \cdot T_c+1}, \ldots, \mathbf{x}_{t-k}, \ldots, \mathbf{x}_t; \\ &\quad \boldsymbol{\sigma}_{t-\alpha \cdot T_c+1}, \ldots, \boldsymbol{\sigma}_{t-m}, \ldots, \boldsymbol{\sigma}_t\big)\end{aligned} \tag{1}$$

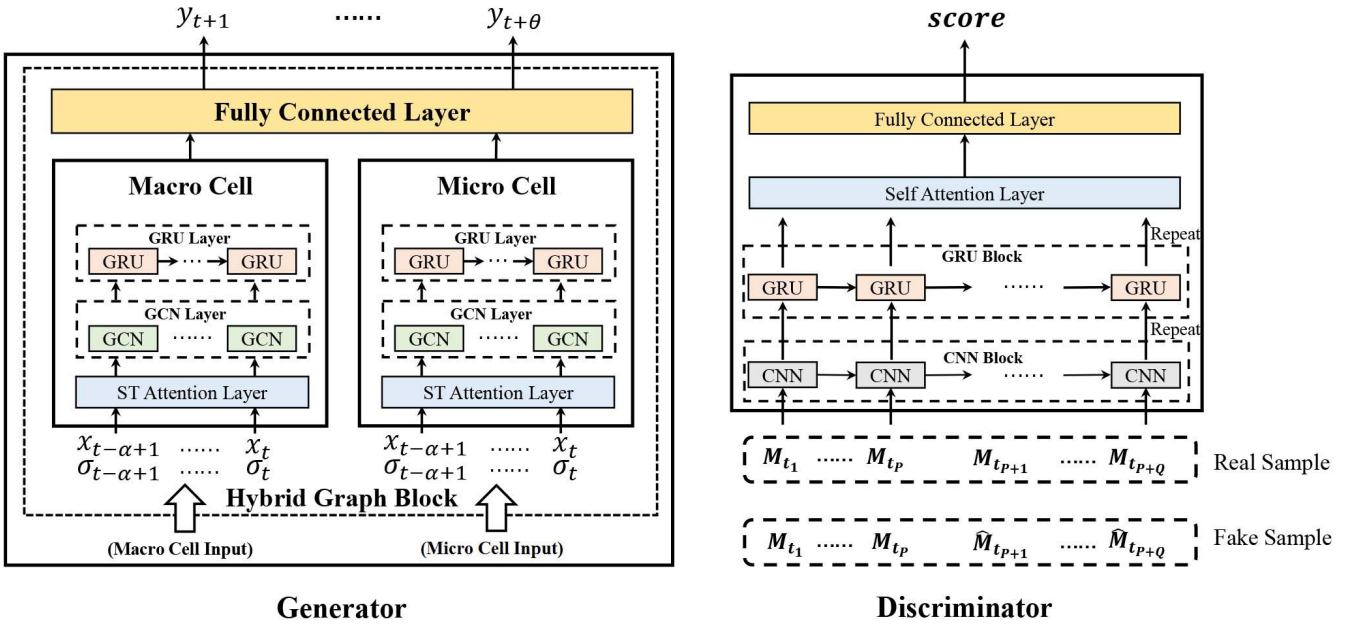


Fig. 2. Structure of the WGAN-based prediction model.

where $F_{dnn}(\cdot)$ denotes deep neural network model. $1 \leq p \leq \theta \cdot T_c$, $0 \leq k < \alpha \cdot T_c$, and $0 \leq m < \alpha \cdot T_c$.

B. Modeling Architecture

Three essential steps are designed for the DNN modeling architecture, including data acquisition, data preprocessing, and prediction. First, historical traffic and other data, including road network information, link traffic state, and signal timing data, are collected into a database. The acquired data is then preprocessed by several procedures: data imputation [25], spatial and temporal alignments, and speed trend clustering. Among them, the purpose of using speed trend clustering is to divide all historical data into several clusters (datasets), and each dataset has its unique feature in the spatial-temporal pattern. Then, data in each cluster is applied for predicting short-term link speed using the hybrid GCN model.

Figure 2 presents the link traffic prediction model based on the WGAN framework, which is driven by the integration of a generator (i.e., a generative neural network) and a discriminator (i.e., a discriminative neural network). Here, the additional conditioning information, feeding into the generator, refers to historical traffic patterns including link traffic data and intersection traffic signal data. The model trains the generator to create predicted link speeds and the discriminator believes they are the ground truth. This study trains the generator and discriminator simultaneously, while model parameters are adapted through the optimization process of the min-max game [24]. The loss function is referred to in the literature in detail. After the GAN-based model is trained by data, the generator is applied to predict link traffic speed.

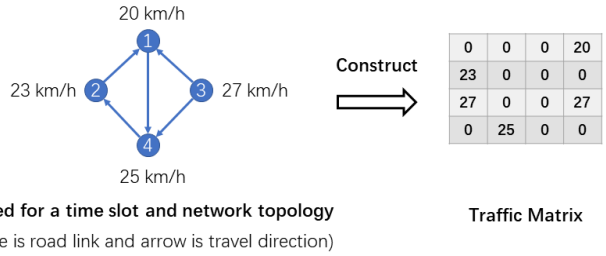
The generative neural network is designed with a hybrid graph block including a micro-modeling cell and a macro-modeling cell. The two types of cells take different input data. The micro-modeling cell uses intersection-wise data to capture effects within the intersection. Whereas, the macro-modeling cell utilizes network-wide data to model the influences of adjacent intersections. Each cell includes three technical components: spatial-temporal attention mechanism (STA), graph convolutional network (GCN), and recurrent neural network (RNN). The STA adjusts the importance of different temporal and spatial contexts. The GCN learns complex topological structures to capture spatial relationships, and the RNN learns temporal changes in traffic data. The modeling details of the generative neural network are presented later in the section.

C. Speed Pattern Clustering

Before applying the GAN-based model, a clustering algorithm classifies link traffic data into different clusters, each of which is considered to have a unique speed pattern. Data in each cluster is then employed for building a link traffic prediction model using the proposed WGAN-based framework. Generally, a clustering algorithm determines different clusters by maximizing both the similarity of the observations within a cluster as well as the dissimilarity of the observations from other clusters. Our study tested three commonly used clustering models, i.e., time-series k-means [26], k-shape, and fast global alignment kernels with k-means [27]. For the time-series k-means method, we also tested three distance measures, and they are Euclidean, dynamic time warping (DTW), and soft-DTW [28].

To evaluate the performance of clustering models, we apply three metrics: Calinski-Harabasz (CH), Davies–Bouldin (DB), and Silhouette Coefficient (SC) score [29]. A higher value of CH indicates modeling results with better cluster assignments. The CH index is represented by the ratio of variances of between-cluster dispersion and within-cluster dispersion:

$$CH = \frac{\sum_i |\mathcal{C}_i| d^2(\mathbf{c}_i, \mathbf{c})/(N_c - 1)}{\sum_{i \in \{1,\cdots,N_c\}} \sum_{\mathbf{x} \in \mathcal{C}_i} d^2(\mathbf{x}, \mathbf{c}_i)/(|\mathcal{D}| - N_c)}, \quad (2)$$



0	0	0	20
23	0	0	0
27	0	0	27
0	25	0	0

Fig. 3. An example of construction operation for traffic matrix.

where $\mathcal{D}$ denotes the whole dataset and N_c denotes the number of clusters. $\mathcal{C}_i$ refers to the set of the i^{th} cluster. $\mathbf{c}$ and $\mathbf{c}_i$ represent the center of $\mathcal{D}$ and $\mathcal{C}_i$, respectively. $d(\cdot)$ refers to a distance function between two vectors. A lower DB value is related to a model with a better disjunction between the clusters and the score is calculated by

$$DB = \frac{1}{N_c}\sum_{i}\max_{j,j\neq i}\left\{\left[\frac{1}{|\mathcal{C}_i|}\sum_{\mathbf{x}\in\mathcal{C}_i} d(\mathbf{x},\mathbf{c}_i) + \frac{1}{|\mathcal{C}_j|}\sum_{\mathbf{x}\in\mathcal{C}_j} d(\mathbf{x},\mathbf{c}_j)\right]/d(\mathbf{c}_i,\mathbf{c}_j)\right\}. \tag{3}$$

A higher SC score indicates that data samples in the same cluster are as close as possible to each other. Whereas, data samples belonging to different clusters are as distant as possible to each other by the following equation:

$$SC = \frac{1}{N_c}\sum_{i}\sum_{\mathbf{x}\in\mathcal{C}_i}\frac{p(\mathbf{x},i)-q(\mathbf{x},i)}{|\mathcal{C}_i|\max\{p(\mathbf{x},i),q(\mathbf{x},i)\}}, \tag{4}$$

where $p(\mathbf{x},i)$ and $q(\mathbf{x},i)$ are calculated by

$$p(\mathbf{x},i) = \frac{1}{|\mathcal{C}_i|-1}\sum_{\mathbf{y}\in\mathcal{C}_i,\mathbf{x}\neq\mathbf{x}'} d(\mathbf{x},\mathbf{x}'), \tag{5}$$

$$q(\mathbf{x},i) = \min_{j,j\neq i}\frac{1}{n_j}\sum_{\mathbf{x}'\in\mathcal{C}_j} d(\mathbf{x},\mathbf{x}'). \tag{6}$$

D. WGAN Framework

Through an aggregation of spatial and temporal traffic information at the network and intersection levels, the aggregated traffic states are described with a macro-model. Traffic characteristics at individual intersections are captured by a micro-model. After getting processed by several hybrid graph blocks, the output features from both macro and micro generative networks are integrated for the final prediction via a fully connected layer. The feature integration is carried out for individual intersections. That is, the hidden state information from the macro-model is fused with the hidden state information from the micro-model for the same road intersection. The fusion operation, carried out by the fully-connected layer, is denoted as

$$\mathbf{f} = f_c(\mathbf{h}_{macro},\mathbf{h}_{micro},\mathbf{w}_{macro},\mathbf{w}_{micro},\mathbf{b}_c), \tag{7}$$

where $f_c(\cdot)$ denotes the function operated by the fully-connected layer, and $\mathbf{h}_{macro}$ and $\mathbf{h}_{micro}$ are hidden states of output feature vectors for the macro- and micro-models, respectively. $\mathbf{w}_{macro}$; $\mathbf{w}_{micro}$, and $\mathbf{b}_c$ are trainable parameters.

In the machine learning process, the link traffic speed predicted by the generator is not directly employed by the discriminator as the input. Instead, inspired by TrafficGAN [30], a converting operation is performed to incorporate link traffic speed at each time into a traffic matrix. To construct a traffic matrix, instant link speeds for the entire network as well as network topology are required. A simple example is illustrated by Figure 3 to show how to represent link traffic speeds of a road network of 4 links by a traffic matrix. Essentially, given a road network with N links, a traffic matrix of $M^{N\times N}$ is produced. The entry of $M_{i,j}$ is represented by 0 if link i is not connected to link j of upstream, or the traffic speed of link i if link i is connected to link j. In this example, $M_{2,1}$ is 23, the speed of link No. 2. Traffic matrix incorporates more information than link traffic speed (i.e., network topology and travel direction), and therefore leads to superior performance in the operation of the discriminator.

Real and fake sample can be compiled for the discriminator using traffic matrices at different times:

$$Sample^{real} = \left\{M_1^{real}, M_2^{real}, \ldots, M_T^{real}, M_{T+1}^{real}\right\} \tag{8}$$

$$Sample^{fake} = \left\{M_1^{real}, M_2^{real}, \ldots, M_T^{real}, M_{T+1}^{pred}\right\} \tag{9}$$

and the only difference between real and fake samples is the last element: the corresponding matrices are generated from real or predicted link traffic speeds, respectively. The Discriminator adopts either real or fake sample data, and each slice of M_t is first processed by a convolutional neural network (CNN) and then a gated recurrent unit (GRU). The output is calculated through a weighted sum process:

$$O = \sum_{t=1}^{T+1} w_t \cdot O_t \tag{10}$$

where w_t is the weight and can be trained via a self-attention mechanism [3], and final hidden representation O is forwarded to a fully connected layer (FC) to produce a score which indicates if the sample is real or fake. Besides, the generator has three technical component units, i.e., spatial-temporal attention (STA), graph convolution network (GCN), and recurrent neural network (RNN), to be described in the following texts.

1) Spatial-Temporal Attention: The idea of STA is to assign weights for input data to provide upper-level representations more accurately [31]. Let us consider that, for each neural network layer, we have an input data block $X \in \mathcal{R}^{N\times C\times T}$, where N is the spatial dimension denoting the number of nodes, C is the number of input feature channels, and T is the length of the temporal dimension. A spatial attentional matrix, A_s, is used to compute a weighted sum of the input data block:

$$A_s = W_s\sigma[(X\mathbf{w}_t)W_{c,t}(\mathbf{w}_c X)^T + B_s]. \tag{11}$$

where σ is the sigmoid activation function, $\mathbf{w}_t \in \mathcal{R}^T$, $\mathbf{W}_{c,t} \in \mathcal{R}^{C\times T}$, $\mathbf{w}_c \in \mathcal{R}^C$, $W_s, B_s \in \mathcal{R}^{N\times N}$ are trainable parameters within the spatial attention mechanism. Similarly, a temporal attentional modulation matrix is defined by

$$A_t = W_t\sigma[(X^T\mathbf{w}_n)W_{c,n}(\mathbf{w}_c' X)^T + B_t] \tag{12}$$

where $W_t, B_t \in \mathcal{R}^{T\times T}$, $\mathbf{w}_n \in \mathcal{R}^N$, $W_{c,n} \in \mathcal{R}^{C\times N}$, and $\mathbf{w}'_c \in \mathcal{R}^C$ are the trainable parameters for the temporal attention module. Both A_s and A_t are further normalized by the softmax function to ensure the sum of attention weights for each node is equal to one. The spatial attentional matrix is accompanied by the adjacency matrix J to dynamically adjust the impacting weights of the graph link between nodes, and the normalized temporal attention matrix is directly applied to the input data block X to generate new input data that captures more cohesive temporal correlations.

2) Graph Convolution Network: The input data, manipulated by STA is further fed into GCN to further capture spatial and temporal correlations embedded in the neighboring nodes of the input data. The graph convolution in the spatial dimension is operated according to spectral graph theory, which generates graph properties by analyzing the Laplacian matrix and its eigenvalues [32]. The Laplacian matrix of a graph is

$$L = D - J, \tag{13}$$

where D denotes the degree matrix and J is the adjacent matrix; the eigen-decomposition of L can be represented by

$$L = U\Lambda U^T, \tag{14}$$

where U and Λ are orthogonal and diagonal matrices respectively. Following the principle of GCN, each node is updated by

$$X_{spa} = Up_\theta(\Lambda)U^T X \tag{15}$$

where $X_{spa} \in \mathcal{R}^{N\times C\times T}$ is the input data of the subsequent layer. $p_\theta(\cdot)$ is a polynomial kernel function.

Let us take the temporal convolutional operation as an example. A convolution layer in the temporal dimension is further stacked to update the signal of a node by merging the information of the neighboring time slices:

$$X_{tem} = \sigma_{ReLU}(f_{CNN}(\Theta, ReLU(X_s))), \tag{16}$$

where $\sigma_{ReLU}(\cdot)$ is the ReLU activation function, Θ is the parameters of the temporal dimension convolution kernel. $f_{CNN}(\cdot)$ denotes the CNN operation and its details are presented in [33]. Similar manipulation in the spatial dimension is also applied.

3) Recurrent Neural Network: An RNN model uses loops in its internal memory to deal with sequential data. Within its structure, a hidden layer receives an input vector from the GCN's temporal convolutional operation and generates the output vector. At each time iteration t, a hidden state $\mathbf{h}_t$ is maintained and updated based on traffic input $\mathbf{x}_t$ and previous hidden state $\mathbf{h}_{t-1}$:

$$\mathbf{h}_t = \mathrm{RNN}_{\sigma_h}(X_t, \mathbf{h}_{t-1}, W_{rnn,x}, W_{rnn,h}), \tag{17}$$

where $\mathrm{RNN}_{\sigma_h}(\cdot)$ denotes the RNN's update functions for hidden states, and $W_{rnn,x}$ and $W_{rnn,h}$ represent weights for the hidden layer. The detailed updating principle of RNN is illustrated in [34].

IV. Case Study

A. Experiment Setup

At a higher level, this paper promotes a general parallel learning framework for the urban traffic control system. A predictive learning approach is presented as an essential component of the framework and is also demonstrated in this section. While the other two components (prescriptive and descriptive learning) have been applied in three other studies [5], [6], [9]. A behavior-cloning-based model was applied to mimic traffic engineers' behaviors when handling urban traffic control operations in the concept of descriptive learning [5]. Thereafter, a large set of traffic control operation data can be generated by the behavior-cloning model. Then, using the traffic control operation data, a predictive model explores the relationships between traffic state and traffic control during the predictive learning process. Based on the predictive model, intelligent control models provide optimal control policy considering the control consequences [6], [9] according to the perspective learning process. Note that the prediction model was evaluated using real-world data collected to justify its effectiveness in the experiment.

The FCD data measured through AutoNavi, a third-party navigation system, was applied to derive link traffic speed, defined as the average speed for vehicles driving on the associated link during a certain time interval. The estimated link traffic speed was then aggregated according to the corresponding signalized cycle of the associated intersection. Signal timing information is obtained from the deployed SCATS system [35], which is an adaptive signal control system capable of adjusting signal parameters every cycle. The case study network, shown in Figure 4, consists of more than a hundred intersections and hundreds of corresponding links. All computational experiments were conducted on a computer server with 32 Intel E5-2620 CPU processors, 4 GeForce RTX 2080 Ti GPU processors, and 64 GB memory.

In our experiments, nodes and their features are respectively defined for macro- and micro-graph models as described in the early section. In particular, a link is modeled as a graph node in the macro-graph, and it has three types of speed statistics as features, i.e.,

$$\begin{aligned}&\mathbf{x}_l^{macro}(t)\\ &= [v_l^{avg}(t), v_l^{max}(t), v_l^{min}(t), v_{agg,l}^{avg}(t), v_{agg,l}^{max}(t), v_{agg,l}^{min}(t)],\end{aligned} \tag{18}$$

where $v_l^{avg}(t)$, $v_l^{max}(t)$, and $v_l^{min}(t)$ denotes average speed, maximum speed, and minimum speed of the link l within a time interval (normally 15 minutes) respectively; $v_{mean,l}^{avg}(t)$, $v_{mean,l}^{max}(t)$, and $v_{mean,l}^{min}(t)$ are the aggregated mean value of the average speed, maximum speed, and minimum speed of the same link and time interval t using data of a longer period e.g. 3 weeks in our case.

A micro-graph model is constructed for each road intersection to describe topological connections between lanes within the same intersection. A node is used to represent a lane, and available turning movement determines the connectivity between lanes. The features of a node contain not only lane

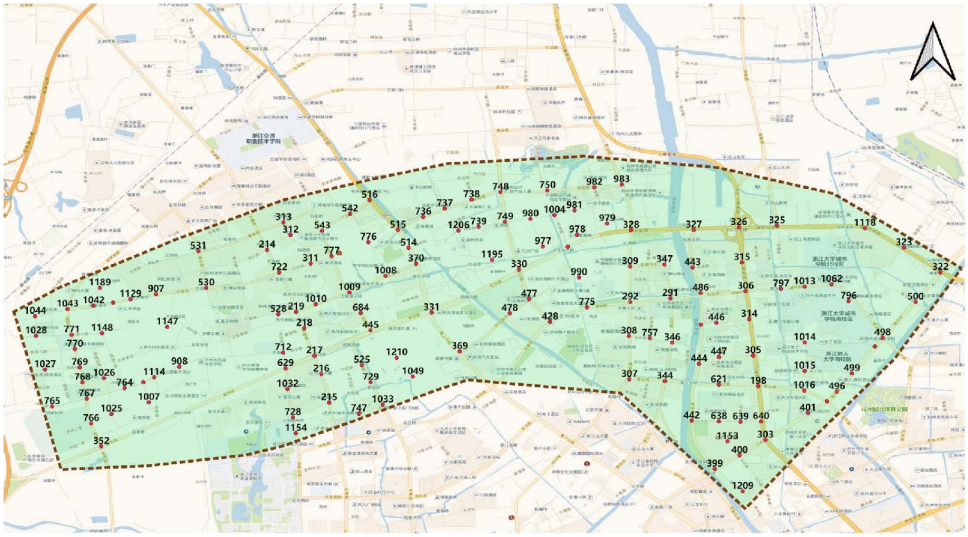


Fig. 4. The signalized road network for the case study in Hangzhou.

speed statistics but also leverage traffic signal information. The time interval for a micro-graph was determined by signal cycle, and in total 30-cycle data were used for constructing the model input $\mathbf{x}_{micro}$. The features of the micro-model are

$$\mathbf{x}^{\alpha}_{micro}(t) = [v^{\alpha}_{avg}(t), v^{\alpha}_{max}(t), v^{\alpha}_{min}(t), v^{\alpha}_{avg,agg}(t), v^{\alpha}_{max,agg}(t), v^{\alpha}_{min,agg}(t), g^{\alpha}_{increase}(t), g^{\alpha}_{decrease}(t)], \tag{19}$$

where $v^{\alpha}_{avg}(t)$, $v^{\alpha}_{max}(t)$, and $v^{\alpha}_{min}(t)$ denotes average speed, maximum speed and minimum speed of lane α at time t, respectively. Similar to the macro model, $v^{\alpha}_{avg,agg}(t)$, $v^{\alpha}_{max,agg}(t)$, and $v^{\alpha}_{min,agg}(t)$ are the aggregated information for lane α. $g^{\alpha}_{increase}(t)$ and $g^{\alpha}_{decrease}(t)$ are two dummy variables indicating whether green signal time is increased and decreased for lane α.

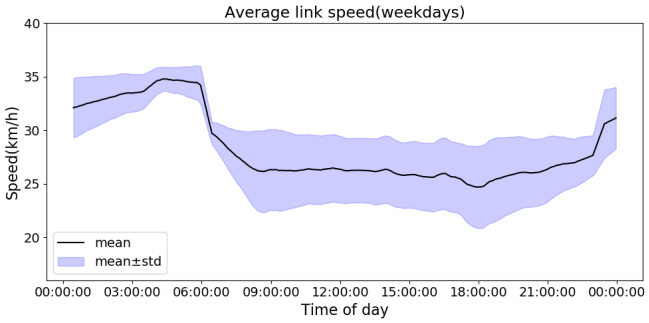
Four-month traffic data from September to December 2019 with a time interval of 3 minutes were applied in our experiments. In this study, the split ratios of training, validation, and test data were set to be 80%, 10%, and 10%. The training process was guided by a method called professor forcing [36], which uses the ground truth from a previous time step as input. The training process applies adversarial adaptation to regulate hidden states in the training and predicting phases to quickly and efficiently train the GAN-based prediction model. The hyperparameters are tuned by an efficient hyperparameter optimization framework called ray tune [37]. For both macro- and micro-models, the kernel sizes of the temporal and spatial convolution are both set to be 1×3 with a stride of 1×1. All the graph convolution layers in each hybrid graph block have 64 output channels (filters), and three hybrid graph blocks were incorporated in the generative network. The discriminator adopts a four-layer CNN (with the kernel sizes all being 2), and a single-layer single-direction GRU, that has a hidden dimension of 64.

B. Data Analysis

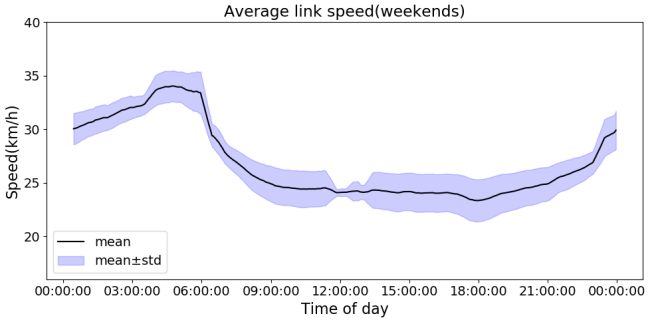
Both link traffic state and intersection signal timing data were analyzed to extract spatial and temporal-variant datasets for training the GAN-based prediction model. Figure 5 shows the change of average link speed and green duration within a day. Each speed sample is estimated using a 30-minute rolling mean window. In general, the link speed on average behaves similarly to each other on weekdays and weekends. Specifically, the speed gradually rises from 00:00 to reach a maximum at around 04:00 and the high speed keeps until 06:00 in the morning. Then, the link speed falls significantly since 06:30 when the morning peak starts. The speed keeps decreasing until 09:00. A relatively low-speed profile is maintained between 09:00 and 17:00. Average link speed degrades from 17:30 and reaches the lowest value at 18:00. Thereafter, the speed progressively increases to above 30 km/h. The traffic of road links behaves similarly during the night of a weekday, from 22:30 to 06:00, and has limited fluctuation on average speed level. The speed patterns of the weekend do not vary significantly from weekdays.

As adaptive signal controllers are deployed in this region, the green durations of all traffic lights simultaneously change in accordance with the fluctuation of the corresponding traffic flows. As shown in Figure 5(c) and (d), the average green duration reaches local maximum values around 08:30 of the morning peak and 18:00 of the evening peak. This is consistent with the variation of traffic demand during the day. Traffic light duration varies more during weekdays, compared to weekends, which indicates the potential spatial variation of traffic flow in the studied region.

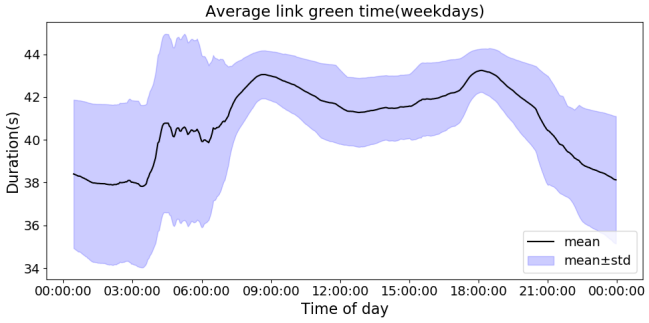
Figure 6 presents the characteristics of spatial-temporal speed patterns on average from 06:30 to 09:00 on weekdays for five neighboring intersections on a selected road, Xiyuan street. Four-month data were used to compute the average link speeds in this plot. The link speed of the southbound direction is generally smaller than that in the opposite direction. A common lag of speed reduction can be observed during the morning peak due to traffic congestion propagation from downstream intersections. Additionally, as shown in Figure 6(b), a regular increase in incoming traffic demand for the northbound direction occurs at around 07:30, causing a progressively decrease in link speed. Comparing the two



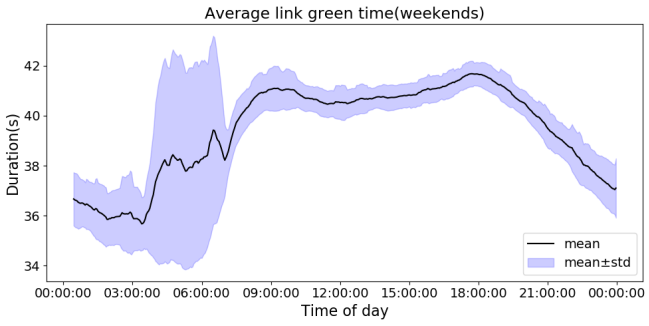
(a)



(b)



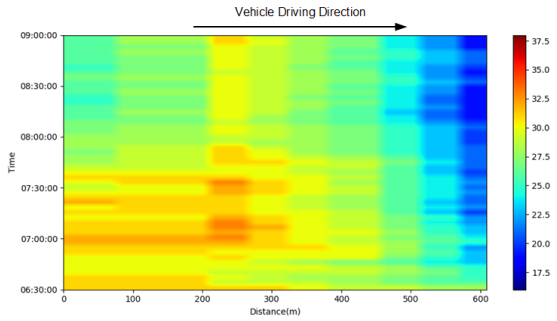
(c)



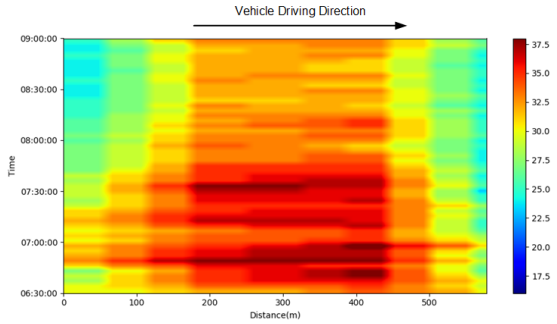
(d)

Fig. 5. (a), (b): Average link speed with a 30-minute rolling mean window for all links in the studied network on both weekdays and weekends from September to December 2019. (c),(d): Average link green time with a 30-minute rolling mean window for all links in the studied network on both weekdays and weekends.

plots therein, the variation in traffic patterns indicates that the link traffic state associated with signalized intersection is affected by its neighboring road links, both upstream and



(a) North to south direction



(b) South to North direction

Fig. 6. (a), (b): Heatmaps of average speed associated with the Xiyuan Road from September to December, 2019 on weekdays for both north/south to south/north directions.

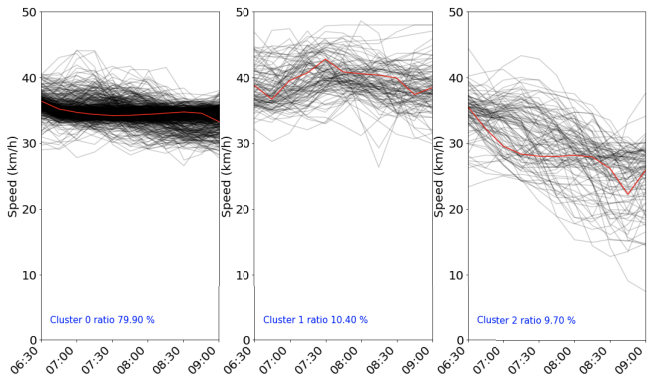


Fig. 7. The clustering results for average link speeds between 06:30 and 09:00.

downstream. Such nonlinear temporal and spatial correlations and dependencies indicate potential challenges for real-time traffic prediction for urban road networks.

C. Link Speed Analysis

Figure 7 shows an example of the clustering result of a set of link speed time series for the morning peak between 06:30 AM and 09:00 AM. All-time series start with a speed of around 40 km/h at 06:30 AM, evolve towards different directions, and then go down towards 20 km/h. The number of clusters is three, meaning that three distinct spatial-temporal patterns are identified for the experimental network. The proportions of the samples in these clusters are 79.90%, 10.40%, and 9.70%.

TABLE I
THE ANALYSIS OF THE BEST CLUSTER MODEL FOR LINK SPEEDS ASSOCIATED WITH EACH TYPICAL PERIODS ON WEEKDAYS

Period	Cluster method	Distance method	Cluster number	CH	DB	SC
00:00-06:30	K-means	Euclidean	3	497.21	0.92	0.60
06:30-09:00	K-means	Soft-DTW	3	667.25	0.82	0.59
09:00-17:30	GAK	–	3	881.99	0.73	0.55
17:30-19:00	K-means	Euclidean	4	1202.21	0.77	0.55
19:00-24:00	K-means	Euclidean	3	830.13	0.81	0.50

TABLE II
PREDICTION PERFORMANCE COMPARISONS FOR DIFFERENT MODELS. (THE BEST RESULTS ARE SHOWN IN BOLD)

Evaluation Metric	HA*	ARIMA*	SVR*	GRU*	GCN*	T-GCN	ST-GCN*	A3T-GCN	PL-WGAN
MAE	8.609	7.976	8.317	7.271	6.756	4.531	5.325	4.512	**4.427**
RMSE	9.373	7.525	9.137	7.989	8.437	6.203	7.114	6.219	**6.154**
ACC	0.614	0.657	0.624	0.681	0.635	0.732	0.702	0.736	**0.747**
R^2	0.435	0.455	0.439	0.447	0.429	0.477	0.461	0.482	**0.495**
VAR	0.424	0.446	0.433	0.451	0.441	0.480	0.468	0.483	**0.488**
Training Time (Seconds)	27	131	1641	3140	2788	5588	7851	8532	11012
Testing Time (Seconds)	4	21	77	53	29	48	115	104	26

* The models include intersection traffic control data for link traffic speed prediction.

The distinct periods for feature clustering are determined by observing local minimums and maximums of average link speed in Figure 5. Regarding each period, 1, 000 samples are randomly selected and such a selection is carried out several times to determine the best cluster model. After tuning the hyperparameters for different cluster models, especially the parameter denoting the number of clusters k, the evaluation results for each period are summarized in Table I. Specifically, traffic patterns for the morning and afternoon peaks are relatively better identified by the same clustering models but with different cluster numbers.

D. Evaluation Metrics

The prediction performance of the model is evaluated using five widely accepted metrics: mean absolute error (MAE), root mean squared error (RMSE), accuracy (ACC), coefficient of determination (R^2), and explained variance score (VAR). Assume T_{tot} and N_l are the total period of the predicted series and the total number of links, respectively. Let $Y_{i,j}$ represent the real link speeds for the i^{th} signalized cycle of T_{tot}. $Y_{i,j}^{pred}$ is the predicted value of $Y_{i,j}$. The metrics are, respectively, computed by the following equations.

$$\text{MAE} = \frac{\sum_{i=1}^{T_{tot}} \sum_{j=1}^{N_l} |Y_{i,j} - Y_{i,j}^{pred}|}{T_{tot} N_l}, \tag{20}$$

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^{T_{tot}} \sum_{j=1}^{N_l} (Y_{i,j} - Y_{i,j}^{pred})^2}{T_{tot} N_l}}, \tag{21}$$

$$\text{ACC} = 1 - \frac{1}{T_{tot} N_l} \sum_{t=1}^{T_{tot}} \sum_{j=1}^{N_l} \left| \frac{Y_{i,j} - Y_{i,j}^{pred}}{Y_{i,j}} \right|, \tag{22}$$

$$\text{R}^2 = 1 - \frac{\sum_{t=1}^{T_{tot}} \sum_{j=1}^{N_l} (Y_{i,j} - Y_{i,j}^{pred})^2}{\sum_{t=1}^{T_{tot}} \sum_{j=1}^{N_l} (Y_{i,j} - \bar{Y})^2}, \tag{23}$$

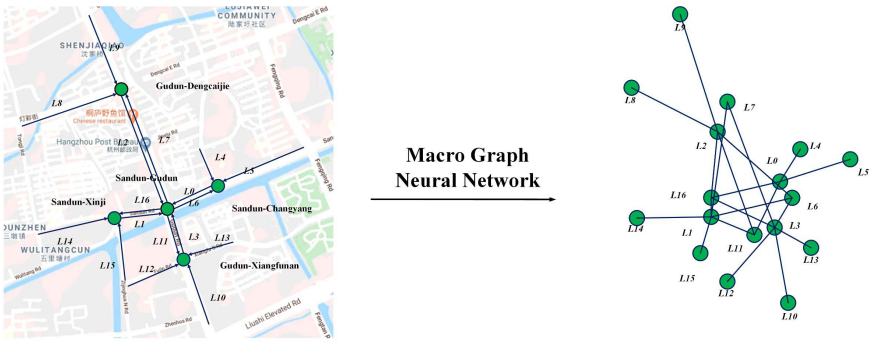
$$\text{VAR} = 1 - \frac{f_{var}(Y - Y^{pred})}{f_{var}(Y)} \tag{24}$$

where $\bar{Y}$ denotes the mean value of Y. $f_{var}(\cdot)$ refers to the variance function.

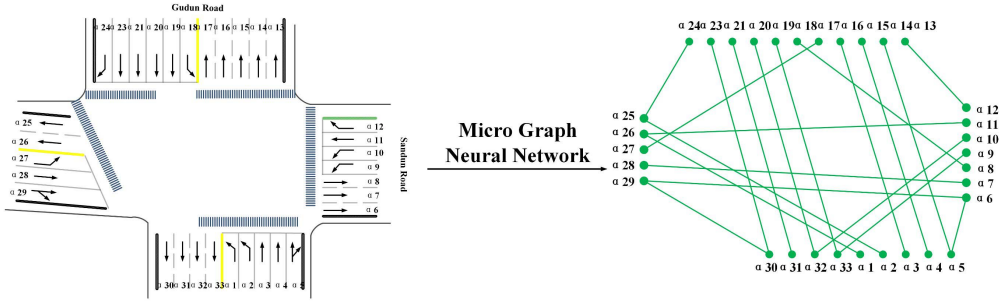
E. Modeling Results and Discussions

In the computational experiment, the performance and effectiveness of the proposed PL-WGAN approach are compared with the eight selected models. The first five are baseline models, and the last three are the state-of-the-art deep learning models that have been recently proposed for traffic prediction. A short illustration of these models is as follows:

- HA: historical average speed calculates the predicted link speeds by taking the mean value of 12 past slices;
- ARIMA: the autoregressive integrated moving average model is a conventional time series model;
- SVR: support vector machine (SVM) model is a popular supervised learning model;
- GRU: gated recurrent unit is a typical sequential model, which specializes in exploring temporal dynamics;
- GCN: graph convolutional network model is a type of graph neural network model capable of capturing structural information from unstructured data [38];
- T-GCN: temporal-graph convolutional network applies GCN and GRU integrated to capture data's spatial and temporal dependence simultaneously [4];



(a) An example for macro graph transformation



(b) An example for micro graph transformation

Fig. 8. The layout and macro and micro graph transformation examples at the Sandun-Gudun intersection.

- ST-GCN: spatial temporal-graph convolutional network applies GCN and gated CNN to capture spatial and temporal dependencies within traffic data [21];
- A3T-GCN: attention-based temporal graph convolutional network follows a similar structure of T-GCN, but the model is extended by an attention mechanism to learn temporal changes of data [3].

To justify the modeling effects of including intersection traffic control, the studied models are divided into two groups. One group includes HA, ARIMA, SVR, GRU, GCN, and ST-GCN models, and intersection traffic control was not considered in these models. Another group includes T-GCN, A3T-GCN, and the proposed PL-WGAN model, which incorporate intersection traffic control data for link speed prediction. The results of Table II show that the models considering traffic control parameters achieve better prediction performance than the alternatives. In addition, the proposed approach PL-WGAN achieves the best prediction performance in terms of selected performance metrics.

Specifically, it ends up with the lowest MAE and RMSE, the highest ACC, R^2, and VAR, among all models. The GAN-based model is benefited from the advantages of the Wasserstein GAN learning strategy and yields a more generalized prediction model that is less prone to overfitting. The computational efforts for training and testing all the models are also demonstrated in the table. Training a GAN-based model is more time-consuming than any other baseline because the model requires dual optimization at the same time. Despite that, the proposed model achieves higher prediction accuracy, in terms of ACC, with a notable margin of 1%. When applied for inferencing, PL-WGAN is also more efficient than the T-GCN and A3T-GCN models (around a fourth of the computational time). Therefore, both prediction and computational performances demonstrate the superiority of PL-WGAN in practical application.

An intersection, Gudun-Sandun, together with its neighboring intersections, Gudun-Dengcaijie, Sandun-Xinji, Gudun-Xiangfunan, and Sandun-Changyang, are constructed into a subset of the macro graph. Figure 8 provides the structure of this part of an example of the macro graph with 17 links while each link is represented by a node and a micro modeling graph transferred from the Gudun-Sandun intersection. Figure 9 further shows the predicted and observed traffic states for the links corresponding to an arbitrarily selected day.

To demonstrate the performance of the model in different traffic conditions, the prediction results of three randomly selected road links (Link 1, Link 7, and Link 9) are shown for different periods of a day (early-peak, off-peak, and late-peak) in Figure 9. These road links exhibit different speed patterns in various periods. For instance, during off-peak (11:00-13:00) and late-peak (17:00-19:00), the speed profiles of all links are relatively stable, although a notable level of fluctuation can be observed during the off-peak (due to decrease traffic flows at noon). During the early peak, it is evident that three road links have different speed patterns, i.e. a gradual decreasing speed for Link 1, and a more fluctuating profile for Link 7 and Link 9. Nevertheless, the proposed model can treat complex situations and achieve relatively accurate predictions in all cases. A small prediction error of less

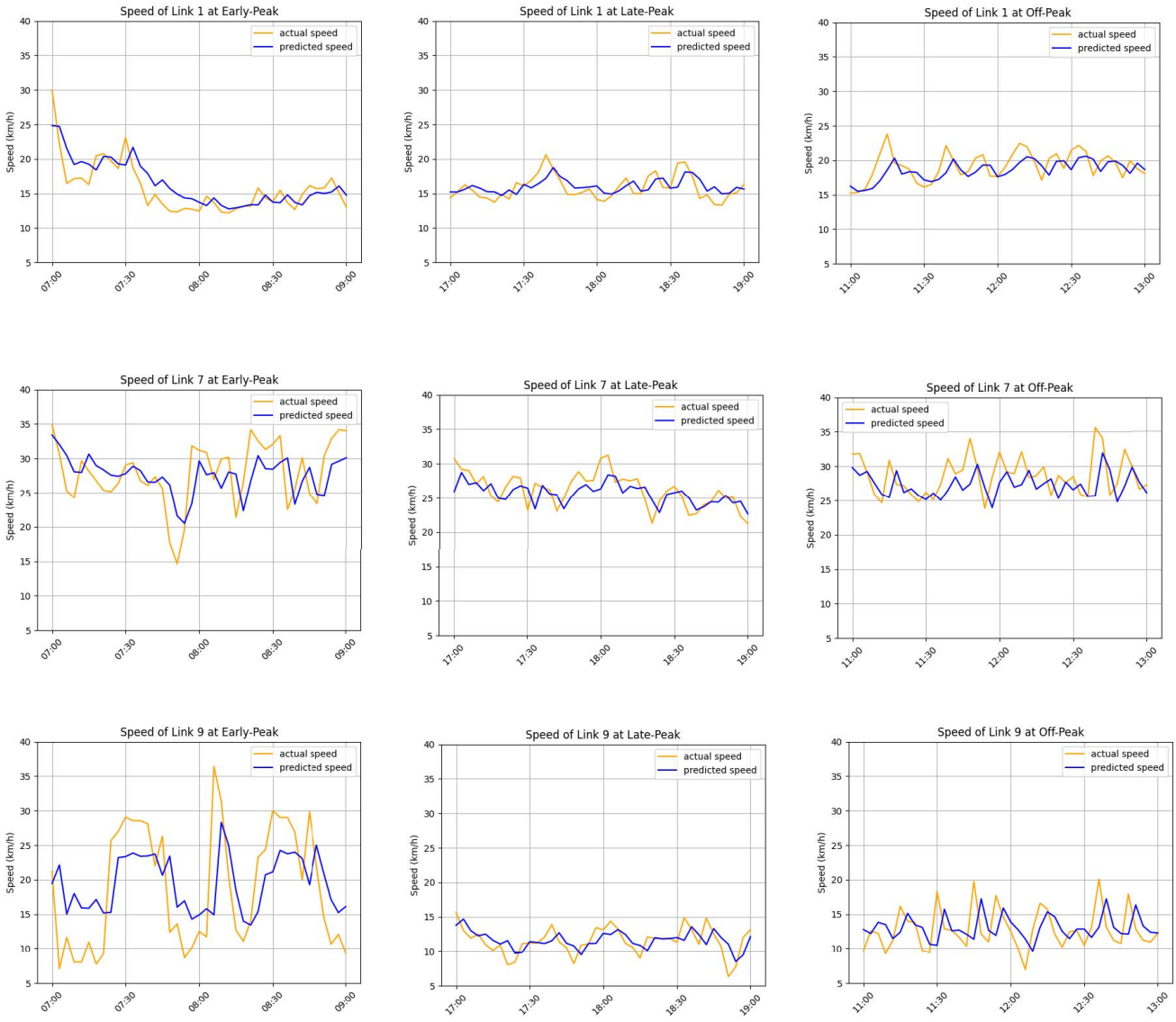


Fig. 9. Real and Predicted Speed for Link 1, 7, and 9 at Early-Peak, Off-Peak, and Late-Peak on November 7, 2019.

than 2 km/h can be maintained for the links most of the time.

V. Conclusion

The revolutionary development of deep learning methods has stimulated the skyrocket of data-driven applications in ITS during the past five years. This study proposes a novel short-term deep learning prediction approach for road traffic in the context of urban road networks. Technically, the GAN-based framework provides a robust approach for training a generalized data-driven model. The model combines a generative neural network and a discriminative neural network. To incorporate spatial and temporal features of road traffic, this paper develops a two-level hybrid GCN model to consider both link traffic characteristics and traffic control factors at intersections. In addition to graph convolution operations, an attention mechanism is applied to further capture temporal heterogeneity and spatial correlation in traffic data. These technical developments together make the model beyond the state of the art of data-driven traffic prediction. Moreover, developing such a learning-based traffic prediction model complements the idea of predictive learning for real traffic, and facilitates the establishment of the PL conceptual framework for traffic control and management.

The proposed model has been implemented and intensively evaluated in a large-scale real urban network in Hangzhou, China. The computational experiments demonstrated the effectiveness and scalability of the model in predicting network traffic state. When compared with several state-of-the-art deep learning models, the proposed model provides the best prediction performance with a limited increase of computation. Further work is to integrate the model for traffic control and management. The idea is to combine the data-driven model with a traffic recommendation system [6] that has been implemented in practical traffic operations in the city. In the longer run, it is promising to further develop and apply the model for traffic management by integrating it with reinforcement learning-based traffic control approaches [9].

REFERENCES

[1] J. Jin *et al.*, "PRECOM: A parallel recommendation engine for control, operations, and management on congested urban traffic networks," *IEEE Trans. Intell. Transp. Syst.*, early access, Apr. 2, 2021, doi: 10.1109/TITS.2021.3068874.

[2] S. Wang, J. Cao, H. Chen, H. Peng, and Z. Huang, "SeqST-GAN: Seq2Seq generative adversarial nets for multi-step urban crowd flow prediction," *ACM Trans. Spatial Algorithms Syst.*, vol. 6, no. 4, pp. 1–24, Aug. 2020.

[3] J. Bai *et al.*, "A3T-GCN: Attention temporal graph convolutional network for traffic forecasting," *ISPRS Int. J. Geo-Inf.*, vol. 10, no. 7, p. 485, Jul. 2021.

[4] L. Zhao *et al.*, "T-GCN: A temporal graph convolutional network for traffic prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3848–3858, Sep. 2019.

[5] X. Li, P. Ye, J. Jin, F. Zhu, and F.-Y. Wang, "Data augmented deep behavioral cloning for urban traffic control operations under a parallel learning framework," *IEEE Trans. Intell. Transp. Syst.*, early access, Jan. 28, 2021, doi: 10.1109/TITS.2020.3048151.

[6] J. Jin, H. Guo, J. Xu, X. Wang, and F.-Y. Wang, "An end-to-end recommendation system for urban traffic controls and management under a parallel learning framework," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 1616–1626, Mar. 2021.

[7] L. Li, Y. Lin, N. Zheng, and F.-Y. Wang, "Parallel learning: A perspective and a framework," *IEEE/CAA J. Autom. Sinica*, vol. 4, no. 3, pp. 389–395, Jul. 2017.

[8] F.-Y. Wang, "Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 630–638, Sep. 2010.

[9] J. Jin and X. Ma, "A multi-objective agent-based control approach with application in intelligent traffic signal system," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3900–3912, Oct. 2019.

[10] Y.-C. Chiu *et al.*, "Dynamic traffic assignment: A primer (transportation research circular e-c153)," Transp. Res. Board, Washington, DC, USA, Tech. Rep. 133232, 2011.

[11] A. Paz and S. Peeta, "Behavior-consistent real-time traffic routing under information provision," *Transp. Res. C, Emerg. Technol.*, vol. 17, no. 6, pp. 642–661, 2009.

[12] Y. Kamarianakis and P. Prastacos, "Space–time modeling of traffic flow," *Comput. Geosci.*, vol. 31, no. 2, pp. 119–133, Mar. 2005.

[13] Y. Wang and M. Papageorgiou, "Real-time freeway traffic state estimation based on extended Kalman filter: A general approach," *Transp. Res. B, Methodol.*, vol. 39, no. 2, pp. 141–167, 2005.

[14] Y. Yuan, J. W. C. van Lint, R. E. Wilson, F. van Wageningen-Kessels, and S. P. Hoogendoorn, "Real-time Lagrangian traffic state estimator for freeways," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 59–70, Mar. 2012.

[15] J. Jin and X. Ma, "A non-parametric Bayesian framework for traffic-state estimation at signalized intersections," *Inf. Sci.*, vol. 498, pp. 21–40, Sep. 2019.

[16] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Dec. 2015.

[17] R. Soua, A. Koesdwiady, and F. Karray, "Big-data-generated traffic flow prediction using deep learning and Dempster-Shafer theory," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2016, pp. 3195–3202.

[18] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang, "Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction," *Sensors*, vol. 17, no. 4, p. 818, 2017.

[19] Z. Cui, R. Ke, Z. Pu, and Y. Wang, "Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction," 2018, *arXiv:1801.02143*.

[20] P. Chen, C. Ding, G. Lu, and Y. Wang, "Short-term traffic states forecasting considering spatial–temporal impact on an urban expressway," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2594, no. 1, pp. 61–72, Jan. 2016.

[21] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 3634–3640.

[22] Z. Cui, L. Lin, Z. Pu, and Y. Wang, "Graph Markov network for traffic forecasting with missing data," *Transp. Res. C, Emerg. Technol.*, vol. 117, Aug. 2020, Art. no. 102671.

[23] Y. Lin, X. Dai, L. Li, and F.-Y. Wang, "Pattern sensitive prediction of traffic flow based on generative adversarial framework," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 2395–2400, Jun. 2019.

[24] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.

[25] Y. Duan, Y. Lv, Y.-L. Liu, and F. Wang, "An efficient realization of deep learning for traffic data imputation," *Transp. Res. C, Emerg. Technol.*, vol. 72, pp. 168–181, Nov. 2016.

[26] X. Huang, Y. Ye, L. Xiong, R. Y. K. Lau, N. Jiang, and S. Wang, "Time series k-means: A new k-means type smooth subspace clustering for time series data," *Inf. Sci.*, vols. 367–368, pp. 1–13, Nov. 2016.

[27] M. Cuturi, "Fast global alignment kernels," in *Proc. 28th Int. Conf. Mach. Learn. (ICML)*, 2011, pp. 929–936.

[28] M. Cuturi and M. Blondel, "Soft-DTW: A differentiable loss function for time-series," in *Proc. 34th Int. Conf. Mach. Learning-*, vol. 70, 2017, pp. 894–903.

[29] Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu, "Understanding of internal clustering validation measures," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2010, pp. 911–916.

[30] Y. Zhang, S. Wang, B. Chen, J. Cao, and Z. Huang, "TrafficGAN: Network-scale deep traffic prediction with generative adversarial nets," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 1, pp. 219–230, Jan. 2021.

[31] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 577–585.

[32] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3844–3852.

[33] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling long- and short-term temporal patterns with deep neural networks," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 95–104.

[34] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, p. 533, 1986.

[35] A. G. Sims and K. W. Dobinson, "The Sydney coordinated adaptive traffic (SCAT) system philosophy and benefits," *IEEE Trans. Veh. Technol.*, vol. VT-29, no. 2, pp. 130–137, May 1980.

[36] A. M. Lamb, A. G. A. P. Goyal, Y. Zhang, S. Zhang, A. C. Courville, and Y. Bengio, "Professor forcing: A new algorithm for training recurrent networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4601–4609.

[37] R. Liaw, E. Liang, R. Nishihara, P. Moritz, J. E. Gonzalez, and I. Stoica, "Tune: A research platform for distributed model selection and training," 2018, *arXiv:1807.05118*.

[38] 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, Aug. 2019.

Junchen Jin (Member, IEEE) received the M.Sc. and Ph.D. degrees in transport science from the KTH Royal Institute of Technology, Stockholm, Sweden, in 2014 and 2018, respectively. He was a Post-Doctoral Researcher with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing. His research interests include intelligent decision-making processes, recommender systems, deep reinforcement learning, simulation and control, and intelligent transport systems.

Dingding Rong received the M.Sc. degree in transport engineering from Shanghai Maritime University, Shanghai, China, in 2017. She is currently a Researcher at the Smart Transportation Research Institute, Enjoyor Company Ltd., Hangzhou, China. Her main research interests include transportation engineering, artificial intelligence, and deep learning.

Tong Zhang received the M.Eng. degree in cartography and GIS from Wuhan University, Wuhan, China, in 2003, and the Ph.D. degree in geography from San Diego State University and the University of California, Santa Barbara, in 2007. He is currently a Professor at the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS), Wuhan University. His current research topics include machine learning, transport geography, and geospatial data analysis.

Qingyuan Ji received the Ph.D. degree in geomatics from Newcastle University, Newcastle, U.K., in 2020. He is currently a Post-Doctoral Researcher with the College of Computer Science and Technology, Zhejiang University, Hangzhou, China. He is also affiliated with Enjoyor Company Ltd., Hangzhou. His research interests include knowledge graph and graph neural networks.

Haifeng Guo (Member, IEEE) received the Ph.D. degree in traffic information and control from Jilin University in 2008. From 2011 to 2013, he was a Post-Doctoral Research Associate with the Zhejiang University of Technology, Hangzhou, China, where he is currently an Associate Professor with the College of Information Engineering. His research interests broadly lie in traffic operation and control, smart transportation systems, and traffic AI.

Yisheng Lv (Senior Member, IEEE) received the B.E. and M.E. degrees from the Harbin Institute of Technology, Harbin, China, in 2005 and 2007, respectively, and the Ph.D. degree in control theory and control engineering from the Chinese Academy of Sciences, Beijing, China, in 2010. He is currently an Associate Professor with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences. His research interests include artificial intelligence, intelligent control, intelligent transportation systems, and parallel traffic management and control systems.

Xiaoliang Ma received the M.Sc. degree in computational science and the Ph.D. degree in transport system from the KTH Royal Institute of Technology, Sweden, in 2001 and 2006, respectively. Before 2013, he worked at the Competence Centre for Traffic Research, a joint research center between KTH, Linköping University, and Swedish Road Transport Research Institute. He has also been an Industrial Consultant on data modeling and intelligent system development. He is currently a Docent in ITS and a Senior Researcher affiliated with the Department of Civil and Architecture Engineering, School of Architecture and Building Environment and Centre of Digital Futures, School of Electrical Engineering and Computer Science, KTH. His research interests include computational modeling and intelligent control of complex systems as well as emerging technologies in ITS.

Fei-Yue Wang (Fellow, IEEE) received the Ph.D. degree in computer and systems engineering from Rensselaer Polytechnic Institute, Troy, NY, USA, in 1990.

In 1990, he joined the University of Arizona, Tucson, AZ, USA, as a Professor and the Director of the Robotics and Automation Laboratory and the Program in Advanced Research for Complex Systems. In 1999, he founded the Intelligent Control and Systems Engineering Center, Institute of Automation, Chinese Academy of Sciences (CAS), Beijing, China, under the support of the Outstanding Overseas Chinese Talents Program from the State Planning Council and 100 Talent Program from CAS. In 2002, he joined the Laboratory of Complex Systems and Intelligence Science, CAS, as the Director, where he was the Vice President for research, education, and academic exchanges at the Institute of Automation from 2006 to 2010. In 2011, he was a State Specially Appointed Expert and the Director of the State Key Laboratory for Management and Control of Complex Systems, Beijing. His current research interests include methods and applications for parallel systems, social computing, parallel intelligence, and knowledge automation. He was elected as a fellow of INCOSE, IFAC, ASME, and AAAS. He was the President of the IEEE ITS Society from 2005 to 2007, the Chinese Association for Science and Technology, USA, in 2005, and the American Zhu Kezhen Education Foundation from 2007 to 2008. He was the Vice President of the ACM China Council from 2010 to 2011 and the Chair of IFAC TC on Economic and Social Systems from 2008 to 2011. He is the President-Elect of the IEEE Council on Radio Frequency Identification (RFID). Since 2008, he has been the Vice President and the Secretary General of the Chinese Association of Automation. He was a recipient of the National Prize in Natural Sciences of China in 2007, the Outstanding Scientist Award from ACM for his research contributions in intelligent control and social computing in 2007, the IEEE Intelligent Transportation Systems (ITS) Outstanding Application and Research Awards in 2009, 2011, and 2015, and the IEEE SMC Norbert Wiener Award in 2014. He was the general chair or the program chair of more than 30 IEEE, INFORMS, ACM, and ASME conferences. He was the Founding Editor-in-Chief of the *International Journal of Intelligent Control and Systems* from 1995 to 2000 and *IEEE Intelligent Transportation Systems Magazine* from 2006 to 2007. He was the Editor-in-Chief of the IEEE INTELLIGENT SYSTEMS from 2009 to 2012 and the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS from 2009 to 2016. He is the Editor-in-Chief of the IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS and the Founding Editor-in-Chief of the IEEE/CAA JOURNAL OF AUTOMATICA SINICA and the *Journal of Command and Control* (Chinese).