

Article

Traffic Flow Online Prediction Based on a Generative Adversarial Network with Multi-Source Data

Tuo Sun ^{1,*} , Bo Sun ^{2,3} , Zehao Jiang ⁴, Ruochen Hao ¹ and Jiemin Xie ⁵ 

¹ Key Laboratory of Road and Traffic Engineering of the Ministry of Education, Tongji University, Shanghai 201804, China; Haoruochen@tongji.edu.cn

² Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hong Kong, China; bobbo.sun@polyu.edu.hk

³ Department of Civil and Environmental Engineering, National University of Singapore, Singapore 117576, Singapore

⁴ Department of Construction Management, School of Civil and Hydraulic Engineering, Huazhong University of Science and Technology, Wuhan 430074, China; jiangzehao@hust.edu.cn

⁵ School of Intelligent Systems Engineering, Sun Yat-sen University, Guangzhou 510275, China; xiejm28@mail.sysu.edu.cn

* Correspondence: suntuo@tongji.edu.cn

Abstract: Traffic prediction is essential for advanced traffic planning, design, management, and network sustainability. Current prediction methods are mostly offline, which fail to capture the real-time variation of traffic flows. This paper establishes a sustainable online generative adversarial network (GAN) by combining bidirectional long short-term memory (BiLSTM) and a convolutional neural network (CNN) as the generative model and discriminative model, respectively, to keep learning with continuous feedback. BiLSTM constantly generates temporal candidate flows based on valuable memory units, and CNN screens out the best spatial prediction by returning the feedback gradient to BiLSTM. Multi-dimensional indicators are selected to map the multi-view fusion local trend for accurate prediction. To balance computing efficiency and accuracy, different batch sizes are pre-tested and allocated to different lanes. The models are trained with rectified adaptive moment estimation (RADAM) by dividing the dataset into the training and testing sets with a rolling time-domain scheme. In comparison with the autoregressive integrated moving average (ARIMA), BiLSTM, generating adversarial network for traffic flow (GAN-TF), and generating adversarial network for non-signal traffic (GAN-NST), the proposed improved generating adversarial network for traffic flow (IGAN-TF) successfully generates more accurate and stable flows and performs better.

Keywords: traffic flow prediction; long short-term memory; convolutional neural network; improved generating adversarial network; rolling time domain; multi-dimensional indicators



Citation: Sun, T.; Sun, B.; Jiang, Z.; Hao, R.; Xie, J. Traffic Flow Online Prediction Based on a Generative Adversarial Network with Multi-Source Data. *Sustainability* **2021**, *13*, 12188. <https://doi.org/10.3390/su132112188>

Academic Editor: Armando Carteni

Received: 7 October 2021

Accepted: 2 November 2021

Published: 4 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As a crucial technology in intelligent transportation systems (ITS) and a research hotspot, traffic flow prediction is necessary for a sustainable traffic network. Traffic flow is determined by modal split and routine choice influenced by many factors, including daily commuting and non-commuting activities, holidays, weather, public activities, traffic events, urban construction, and individual relocation.

In recent years, many deep learning-based models have been proposed to predict traffic volume, and they process the time-series traffic data [1,2]. Because there is no explicit order dependence for traffic estimation, the observations in a period are treated as the same to predict a series of future journey times via the simulation of the historical observations' distribution, which time series methods are not suitable to handle. Therefore, the deep learning method is considered based on generative adversarial networks (GANs) [3]. They are promising image generation methods because they generate samples based on their simulated probability distribution of a complex dataset. In a GAN, which follows an

adversarial process to estimate generative models, two models are trained simultaneously. A generative model determines the distribution of training data, while a discriminative model calculates the probabilities of the sample selections. The training procedure aims at maximizing the likelihood of making a mistake. Like a minimax two-player game, there is a unique solution in the space of arbitrary functions, with the possibility of recovering the training data or discovering fake and real data equal to 0.5 universally. Radford et al. [4] introduced the deep convolutional generative adversarial networks (DCGANs), which are a class of convolutional neural network (CNN) and have certain architectural restraints. DCGANs were proved to be powerful for unsupervised learning. The training results of different image datasets showed the representation level from object part to scene in the generator and discriminator. Mescheder et al. [5,6] analyzed the common training algorithms of a GAN. Using the form of a smooth two-player game, the correlation gradient vector field of the GAN training target was analyzed.

The general GAN structure was proved to be superior in training. Also, they showed the convergence in the well-known GAN structure, which is difficult to train. Arjovsky et al. [7] introduced a new algorithm. They proved that it could enhance learning stability, eliminate pattern failure, and deliver meaningful learning curves, which helps debug and super parameter search. Zhang et al. [8] proposed a travel information maximization generation countermeasure network based on deep learning. By modeling the joint travel time distribution (TTD) of two continuous links and considering the spatiotemporal correlation of the whole network, the TTD was estimated successfully in the framework of a GAN for the first time. While multilayer perceptrons define the generative and adversarial models, the most recent research is based on the long short-term memory (LSTM) or CNN for various applications. LSTM, a fundamental deep learning model, can learn long-term dependencies. An LSTM internal unit, which includes a cell, an input gate, an output gate, and a forget gate, has hidden states improved by nonlinear mechanisms which allow states to propagate without amendment and use simple learned gating functions to update or reset states. LSTM works exceptionally well on different problems, including handwriting recognition, electric load forecasting, and natural language text compression. As a deep, feed-forward artificial neural network class, CNNs have been successfully used to analyze visual imagery. A CNN consists of multiple hidden layers (including convolutional layers, pooling layers, fully connected layers, and normalization layers), but it only has one input layer and one output layer. The applications of CNNs include natural language processing, recommender systems, and image and video recognition. As the traffic data are not preprocessed, most artificial intelligence (AI)-based prediction models, and being single input, cannot match the traffic variation among various intervals (e.g., hours, days, and weeks), which leads to poor performance under different traffic conditions. This paper puts forward a GAN-based prediction model combining BiLSTM as the generative model with a CNN as the discriminative model, in which a rolling time domain scheme is proposed by dividing the dataset into slices of training and test data on the same period to capture the variation of traffic data and realize the self-learning ability of GAN. Additionally, traffic flow prediction considering influence under signal control has generally not been considered in previous works. This paper adopts min-max normalization pre-processing of traffic flow and signal timing data as the input parameters to resolve this issue.

The remainder of this paper is organized as follows: we review the literature on GAN methods in Section 2. Then, the methodology framework of the improved generative adversarial network framework for forecasting traffic flow, including the adopted structure, rolling time-domain training and testing, model fusion mechanism, and multi-dimensional indicators, is presented in Section 3. The experimental dataset, evaluation indexes, distribution probability, model calibration, and time series stationary test process are in Section 4. Experimental results and comparison with benchmark methods are analyzed in Section 5. Section 6 concludes with the contributions and future work of this research.

2. Literature Review

Many existing predicted methods have been suggested to resolve the complex relationship between different factors and traffic fluctuation. One is the statistical-based method, which is simple to calculate and maintains stable prediction accuracy. However, the statistical-based method is required to satisfy linear or other strong assumptions related to statistical properties. Thus, it cannot accurately reflect the complex and nonlinear relationship hidden in time-varying traffic flows. Also, the predicted flows may lose accuracy since the utilized statistic characteristics can eliminate local fluctuations to ensure a relatively acceptable accuracy in general. Another type is the AI-based method, which has strong abilities of multi-source information fusion and nonlinear prediction. Still, it often faces problems of overfitting and calculation costs for online applications. Both of these methods can be applied to offline and online scenarios. However, due to the complex and nonlinear relationship between the influence factors and traffic flows, it is impossible to fit the time-varying properties of traffic flows into a fixed offline expression.

The traditional AI methods for traffic flow prediction can be grouped into two types: parametric and non-parametric methods. The parametric methods include the artificial neural network (ANN), autoregressive integral moving average model (ARIMA), exponential smoothing, and Kalman filter (KF). Voort et al. [9] suggested a hybrid method for short-term traffic prediction, which could better handle traffic flow changes and significantly improve prediction performance; Ding et al. [10] divided traffic patterns into six levels, and a time-space-based ARIMA was established to predict urban traffic flow at 5-min intervals. Chandra and Al-Deek [11] established a fast-path short-time traffic flow vector autoregressive prediction model, which considered the influence of upstream and downstream location information and achieved good results. Huang et al. [12] proposed a hybrid model in which the linear and non-linear parts were processed by the ARIMA and the ANN, respectively. The hybrid model improved the prediction accuracy. KF-based traffic prediction research was proposed because of its recursive solid ability. Gong et al. [13] suggested a KF-based model for the prediction of short-term traffic volume, and they used the Internet of Things for data collection. Ojeda et al. [14] adopted the adaptive scheme to build a multi-step KF-based method for traffic flow prediction. Guo et al. [15] argued that the adaptive KF-based method for the short-term traffic prediction did not have a large flow rate. The short-term traffic flow prediction performance could be improved by a new approach with better adaptability and stability.

Compared to parametric methods, non-parametric methods are more complicated and flexible because of the uncertainty of groups of parameters and traffic data structure. Based on statistical theory, the support vector machine (SVM) is adopted in many areas. However, scholars have established a least squares SVM, chaotic wavelet SVM, particle swarm optimization SVM, and genetic algorithm SVM because of its sensitivity in selecting kernel functions and parameters. Another type of typical non-parametric method is the neural network, which is widely used in prediction [16,17]. The improved extreme gradient boosting (XGBoost) with spatial lag has improved prediction accuracy [18]. Neural networks are good at learning multidimensional complex nonlinear problems. Khotanzad and Sadek [19] used fuzzy neural network and multi-layer sensing to simulate fast-path network, and the result was better than that from the parametric regression model. Qiu et al. [20] built a Bayesian regularized neural network to forecast short-term velocity. Ma et al. [21] constructed a large-scale congestion prediction model and proposed the restricted Boltzmann machine based on a recurrent neural network. In recent years, an increasing interest in deep neural networks, such as deep belief networks [22], has appeared in traffic prediction. Bidirectional long short-term memory (BiLSTM) has an excellent predictive application effect. Cui et al. [23] suggested a deep overlay bidirectional, unidirectional LSTM neural network structure. The forward and backward correlation of time series data were used, and the structure was applied to predict the whole network's traffic speed. With the backward dependence of traffic data for prediction, the model can predict the traffic speed of freeways and complex urban traffic networks. Deep learning-based

GANs can deal with these drawbacks. More hybrid prediction frameworks of parametric and non-parametric methods have been proposed due to the complex and nonlinear relationship among time-varying traffic flow sequences. Through training, a high-quality data confusion discriminant model is generated, which is widely used in image restoration, semantic segmentation, image prediction, etc., because GANs can flexibly perform real and virtual scenes in tasks such as artificial systems, computational experiments, and synchronous execution. Wang et al. [24,25] summarized GAN's latest technology and found that the parallel system research of functional interaction and integration has great potential. Liang et al. [26] considered the traffic state of the original traffic and the traffic state of the training samples based on GAN, and estimated the missing traffic state under spatio-temporal and flow-speed relationships. Li et al. [27] utilized the joint distribution of relative entropy to reduce the difference in different traffic distribution situations and established a multi-quantity based on a GAN, which could improve the prediction performance of a traffic prediction model for extreme values.

Reviewing the above literature, we found three critical problems for online traffic prediction:

1. How to find the periodicity and trend of flows in the time domain?

Though special cases and many other factors influence traffic flows to change frequently, traffic patterns still show an obvious tendency in most cases with stable periodicities and trends. Daily, weekly, monthly, and quarterly patterns of traffic flows are significant fluctuation time domains for prediction approaches.

2. How to describe the local trend raised by special cases with steady accuracy?

There is a correlation among local temporal neighbors. However, the local closing values tend to be heavily fluctuating. Their prediction is unreliable based mainly on closing values. The prediction module should depend on groups of statistic local information from temporal neighbors for steady accuracy. In this way, the fluctuation direction and range can be resolved in a reasonable domain.

3. How to maintain the continuous self-learning ability in the time domain?

As complex factors influence traffic flows, the ability of the prediction model to continuously learn the latest pattern of sudden variations, periodicities, and trends is noteworthy, especially for online prediction methods. In most offline methods, the parameters or strategies are fixed, making it a challenge to deal with unknown fluctuations. Online methods are mostly trained with the latest small dataset with a calculation limitation, which cannot fit long-term periodicity and trends. As a result, an online discriminative tool is necessary for the frequent online prediction process. Recently, the loss of most online methods has usually been focused on the current interval, regardless of the whole sequence discrimination. In urban signalized intersections, signal timings must be considered to improve the self-learning ability of online prediction methods. To enhance traffic prediction, this paper presents an improved generative adversarial network framework for traffic flow (IGAN-TF) which introduces a parametric-vector-based fusion method to fuse three components' closeness. Then, multiple temporal views (i.e., daily periodicity, weekly periodicity, biweekly periodicity, and monthly trend) are employed to different temporal sequences with different periodicities and trends. The method of rolling time-domain training and testing further expanded the use of data. A relative strength index is introduced for the complex and nonlinear relationship between the target traffic flows. Five kinds of indicators are resolved by the generator and the discriminative model of IGAN-TF. Finally, the effectiveness and accuracy of the proposed model are proved by comparing the prediction results of different lanes (straight-lane, left-lane, right-lane, and straight right-lane) at the intersection with those of the baseline methods.

In conclusion, for question 1, time series models with long-term and short-term memory such as LSTM and BiLSTM are selected as basic units in the basic model. At the data fusion level, the model combines daily, weekly, biweekly, and monthly features. In addition, in terms of data feature extraction, trend features media and moving average

are selected to learn traffic flow data trend features and make accurate predictions. For question 2, the model captures the local trend of the data. At the level of data fusion, recent features are combined. In addition, the selection of volatility features' maximum, minimum, and signal timing phase time dramatically impacts the traffic in the data feature extraction. For question 3, the self-learning ability proposed in this paper is mainly through introducing the GAN mechanism, inputting real data, generating data, and constantly judging and resisting the judgment model to realize the final prediction. In this process, the generative judgment model will continuously feedback the loss to the generative model to realize self-learning.

3. Methodology

In this section, we first give the predicted problem definition and motivation to illustrate the time series composition and relationship of traffic data. Then, the proposed improved generating adversarial network for traffic flow (IGAN-TF) is depicted in detail, and the corresponding algorithm (Algorithm 1) and dataset divided strategy are introduced to design advanced models.

3.1. Problem Definition and Motivation

Although traffic flow is simply time-series data, it is widely influenced by various factors, making it difficult to be fitted by a fixed expression without continuously renewing the latest fluctuation information. The travel demand is mainly composed of regular daily commuting and non-commuting active travels, which form a relatively stable traffic pattern. Special cases (such as weather change, public activities, and traffic events) may suddenly change the short-term travel demand. Other factors, such as holidays, urban construction, and individual relocation, could affect the traffic flows in the long term. Unfortunately, these kinds of data are challenging to collect entirely and to fuse. In urban signalized intersections, signal timings are another significant factor that affects traffic flows in each lane. However, no matter how these factors influence the traffic flow, the combined effect of these factors is reflected as the variation of traffic data in the time domain with various fluctuation ranges and temporal ranges as shown in Figure 1.

In Figure 1a, weekdays have much higher peak values than weekends. However, traffic flows fall obviously during relative long-term holidays because the outflow from this studied area outnumbers the inflow to this studied area during holidays. The two curves in Figure 1b have different shapes, which demonstrates that special cases could result in a sudden closing local trend at any time. Figure 1c describes daily traffic flows in a newly constructed road from 6 October 2016 to 31 December 2017. Traffic flows gradually increased with an obvious trend as time passed. Hence, traffic flow online prediction is required to precisely capture the fluctuation range and temporal range by considering different degrees of temporal influence by closeness, periodicity, and trend.

3.2. Improved Generative Adversarial Network Framework for Traffic Flow

With the high-frequency detected traffic flows, high-quality one-step forecasting is generally of great concern to traffic engineers and policymakers to design robust traffic management schemes. To tackle the above problems, an IGAN-TF is proposed to predict traffic flows by combining BiLSTM as the generative model (generator) G with CNN as the discriminative model D . The traffic flow of each lane is predicted in one step with the IGAN-TF, solely based on its historical information. By introducing an adversarial process between the generator G and the discriminative model D , the online discriminative model can judge the generated traffic flows of the generator G continuously, which improves the self-learning ability of online prediction.

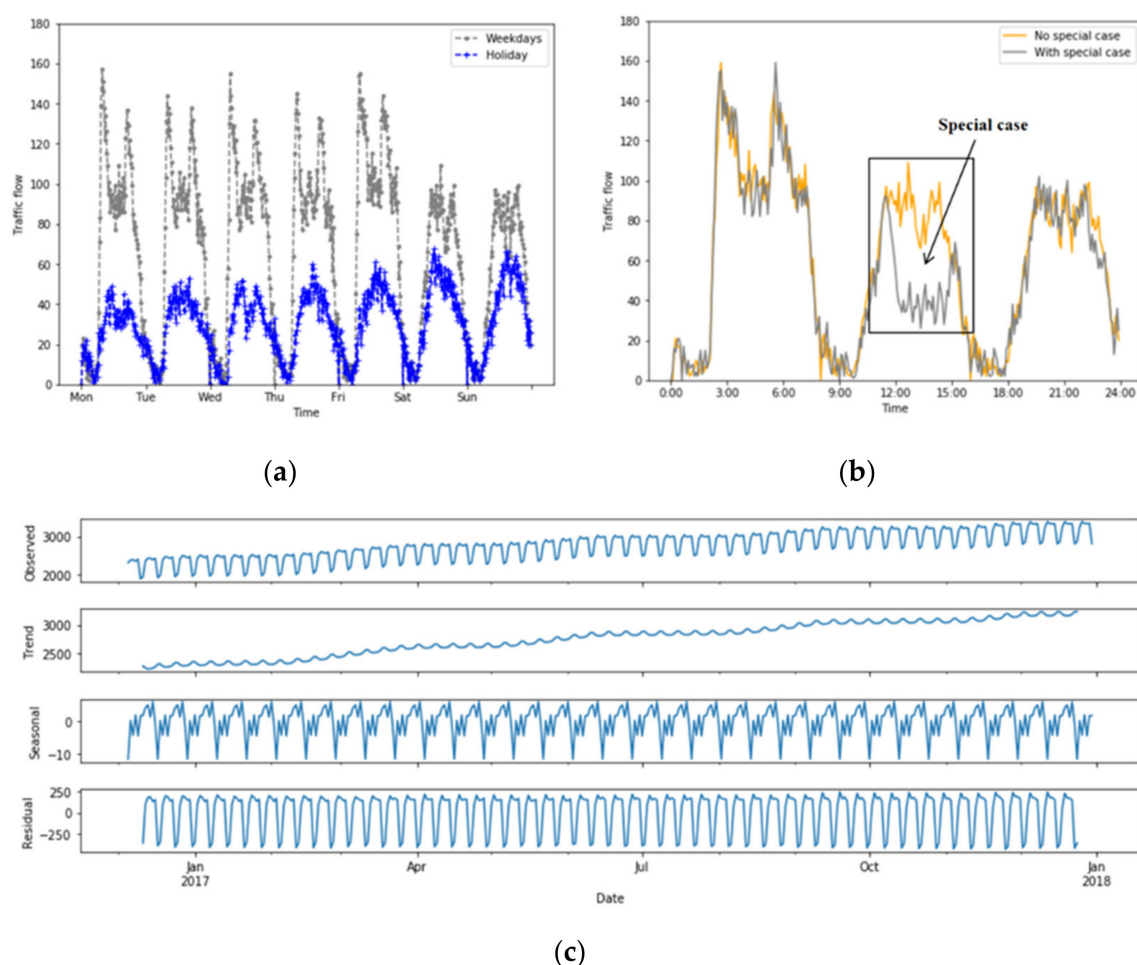


Figure 1. Different influences of factors on time-varying traffic flows. (a) Daily traffic pattern with periodicity and holiday long-term influence. (b) Special case with closing local trend. (c) Factors with long-term trend.

Table 1 presents the notations used in IGAN-TF. Let i_t represent a set of multi-dimensional indicators and f_t denote the closing traffic flow for an l -minute interval at the interval t . Given the past closing S timesteps flows f_1, f_2, \dots, f_S and the corresponding historical multi-source indicators $I_{S+1}(i_1, i_2, \dots, i_S)$, the goal is to predict the closing flow \hat{f}_{S+1} for the following l -minute time interval.

Table 1. Notation.

Notation	Description
T	Total number of intervals in the time domain of the dataset
l	Time length of every detected interval in the dataset (unit: min)
K	Batch_size of the minibatch training process
S	Timestep of BiLSTM, which is the number of hidden units during each training/testing iteration
E	Selected dimension number of multi-dimensional indicators
i_t	Multi-dimensional indicators at the interval $t = 1, 2, \dots, S$, and each dimension represents one indicator
$i_t(n)$	n -th dimensional indicator at the interval t
$I_{S+1}(i_1, i_2, \dots, i_S)$	Multi-dimensional indicators of temporal sequence S to generate target traffic flow at the interval $S+1$
\hat{f}_{S+1}	Predicted target traffic flow at the interval $S+1$, generated from outputs of Generator G
f_t	Traffic flow ground truth at the interval t
$f(f_1, f_2, \dots, f_S, f_{S+1})$	Traffic flow ground truth of temporal sequence $1, 2, \dots, S, S+1$ (f is a vector form)
$\hat{f}(f_1, f_2, \dots, f_S, \hat{f}_{S+1})$	Generated traffic flow of temporal sequence $1, 2, \dots, S, S+1$ (\hat{f} is a vector form)

Figure 2 shows the framework of IGAN-TF. IGAN-TF is composed of two processes, including Figure 2a for data preparation section and Figure 2a for model learning/predicting section as follows.

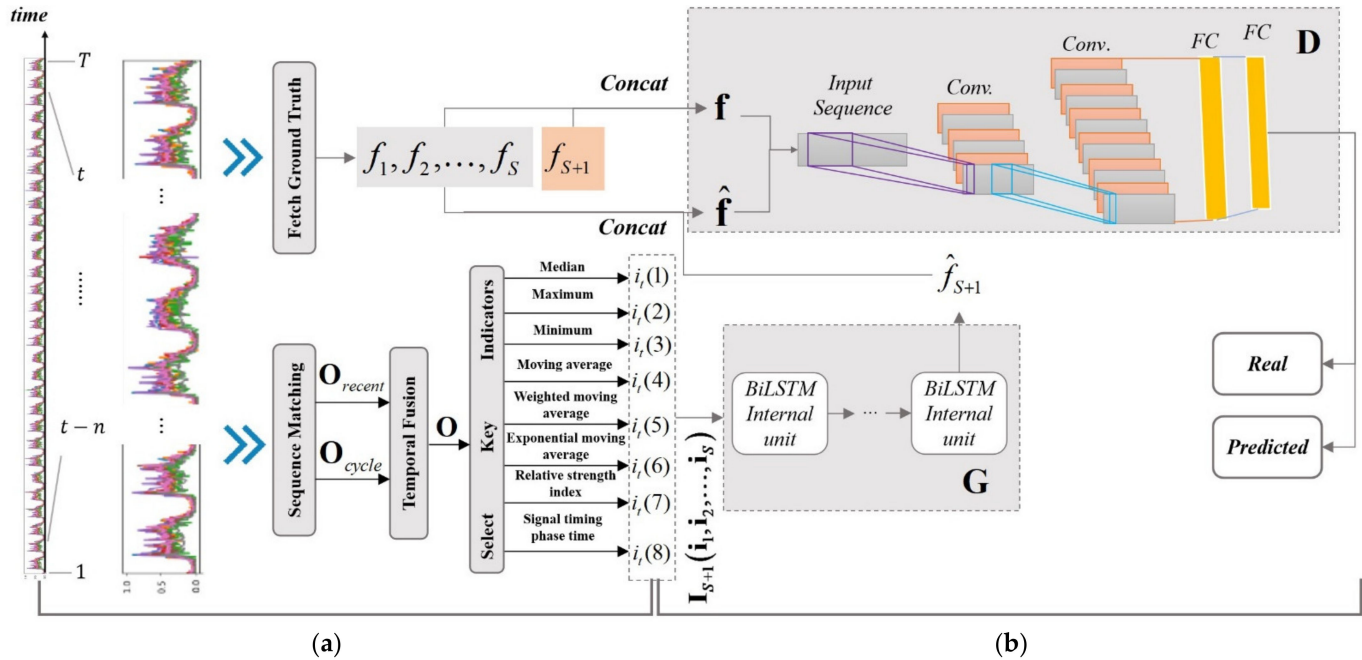


Figure 2. IGAN-TF framework. (a) Data Preparation. (b) Model Learning/Predicting.

(a) Data preparation

In the time domain T , traffic flows in different lanes are firstly transferred into a range $[0, 1]$ with min–max normalization. The ground truth sequence of the targeted interval f_{S+1} and neighbors f_1, f_2, \dots, f_S are concatenated as f and prepared to be input into the discriminative model. Moreover, traffic ground truth in multiple temporal views, such as in the daily, weekly, biweekly, and monthly domain, are matched with the recent intervals by multi-view temporal fusion, respectively, as time goes by. Note that some of the indicators, which are not related to temporal views, are not determined from the multi-view temporal fusion and remain the same value in the current time. With the temporal fusion, E key indicators $i_t(i_t(1), i_t(2), \dots, i_t(E))$ are selected to form the input data of generator G .

(b) Model learning/predicting

Since traffic flow data is a specific time-series data, the BiLSTM model, widely used for long short-term time series prediction, is selected as the generator G to predict output \hat{f}_{S+1} according to the input data $I_{S+1}(i_1, i_2, \dots, i_S)$.

$$\hat{f}_{S+1} = G(I_{S+1}) \quad (1)$$

Based on the CNN architecture, the discriminative model D calculates the probability of whether a sequence comes from the dataset $f(f_1, f_2, \dots, f_S, f_{S+1})$ or is being produced by a generator G with $\hat{f}(f_1, f_2, \dots, f_S, \hat{f}_{S+1})$ by performing convolution operations on the one-dimensional input sequence.

The generator G and the discriminative model D evolve and interact with each other. One side evolves, while the other loses, and finally, the evolution ceases until the false target is similar to the true target. With historical ground truth f_1, f_2, \dots, f_S and indicators $I_{S+1}(i_1, i_2, \dots, i_S)$ as the inputs, the discriminative model D should be as “confused” as possible. The generator G should diminish the adversarial loss because the discriminative model D will not make correct discrimination on the prediction.

Mean absolute error (MAE) training in neural networks may result in missing the minimum point during gradient descent. In contrast, the gradient of mean squared error (MSE) will decrease as the loss decreases. The Huber loss is chosen as the adversarial loss for generator G , which combines the advantages of both and makes the gradient reduce and fall near the minimum. The generator G minimizes $L_G(f, \hat{f})$ while keeping the weights of D fixed.

$$L_G(f, \hat{f}) = \begin{cases} \frac{1}{2}(f - \hat{f})^2 & \text{for } |f - \hat{f}| \leq \delta \\ \delta|f - \hat{f}| - \frac{1}{2}\delta^2 & \text{otherwise} \end{cases} \quad (2)$$

where δ is the Huber loss function threshold.

Since the role of the discriminative model is just to classify the input sequence $f(f_1, f_2, \dots, f_S, f_{S+1})$ into 1 and $\hat{f}(f_1, f_2, \dots, f_S, \hat{f}_{S+1})$ into 0, the discriminative model D aims to minimize the sigmoid cross-entropy loss function L_{sce} , keeping the weights of G fixed.

$$L_D(f, \hat{f}) = L_{sce}(D(\hat{f}), 0) + L_{sce}(D(f), 1) \quad (3)$$

The generative BiLSTM and discriminative CNN are trained iteratively. During the adversarial process, the generator and discriminator are iteratively trained with mini-batches of size K by summarizing the losses over every batch_size during an iteration, accelerating convergence and improving accuracy. Due to the temporal urgency of online prediction, disparate batch sizes are tested in this paper. The state-of-the-art optimizers for parameters, such as mini-batch gradient descent (MBGD), stochastic gradient descent (SGD), adaptive moment estimation (Adam), and momentum, have a series of problems in saddle point, time consumption, and learning rate selection. To further stabilize training, accelerate convergence, and improve generalization, an advanced optimizer RAdam [28] is applied to train IGAN-TF by rectifying the adaptive learning rate to have a consistent variance. The iterative process is summarized below with batch_size K . β_1 and β_2 are decay rates to calculate moving average and moving 2nd moment, respectively.

Algorithm 1 Iterative mini-batch adversarial training with backpropagation and learning rate decayed by rectified Adam as shown.

Step 1: Set the initial learning rates of the generator and the discriminative model (i.e., ρ_G and ρ_D , respectively);

Step 2: Initialize exponential decay moving 1st moment, m_0 , exponential decay moving 2nd moment, v_0 , and weights of the generator and the discriminative model (i.e., W_G and W_D , respectively);

Step 3: Compute the maximum length of the approximated simple moving average (SMA) with $\lambda_\infty = \frac{2}{(1-\beta_2)} - 1$, while $t = 1, \dots, S$ and the convergence is not reached;

Step 4: Train the generator G with RAdam;

- (1) Get a new sequence $(i_1, f_1), (i_2, f_2), (i_3, f_3), \dots, (i_K, f_K)$ for the generator G ;
- (2) Calculate gradients with regard to the stochastic objective at timestep t with $g_t^G = \sum_{k=1}^K \frac{\partial L_G(i_k, f_k)}{\partial W_G}$;
- (3) Update exponential moving 2nd moment with $v_t^G = \beta_2 v_{t-1}^G + (1 - \beta_2)(g_t^G)^2$ and exponential moving 1st moment with $m_t^G = \beta_1 m_{t-1}^G + (1 - \beta_1)g_t^G$;
- (4) Compute bias-corrected moving average with $\hat{m}_t^G = \frac{m_t^G}{(1-\beta_1^t)}$ and the length of the approximated SMA with $\lambda_t^G = \lambda_\infty - \frac{2t\beta_1^t}{(1-\beta_1^t)}$, respectively;
- (5) If $\lambda_t^G > 4$, compute bias-corrected moving 2nd moment $\hat{v}_t^G = \sqrt{\frac{v_t^G}{(1-\beta_2^t)}}$ and the variance rectification term $r_t^G = \sqrt{\frac{(\lambda_t^G - 4)(\lambda_t^G - 2)\lambda_\infty}{(\lambda_\infty - 4)(\lambda_\infty - 2)\lambda_t^G}}$ and update weights with adaptive momentum $W_t^G = W_{t-1}^G - \frac{\rho_G r_t^G \hat{m}_t^G}{\hat{v}_t^G}$; otherwise, update weights with $W_G = W_G - \rho_G \hat{m}_t^G$;

Step 5: Train the discriminative CNN by updating weights W_D with RAdam, and the same with the generator;

Step 6: End the algorithm if reaching convergence.

3.3. Strategies for Improving IGAN-TF

Concerning the three problems mentioned in Section 2, a series of strategies are proposed to improve the performance of IGAN-TF.

3.3.1. To resolve the Periodicity and Trend of Flows in the Time Domain

(a) To resolve the periodicity and trend of flows in the time domain

In terms of cross-validation, the training and testing datasets are treated as a rolling time domain scheme shown in Figure 3 to capture the temporal periodicity and trend of the dataset. In the first cycle, we select the first M days as the training set and the next N days as the testing set; in the second cycle, the $N + 1$ days to $M + N$ days are chosen to form the training set, and $M + N + 1$ days to $M + 2N$ days are selected for testing. The process is repeated until all the data are in the experiment.

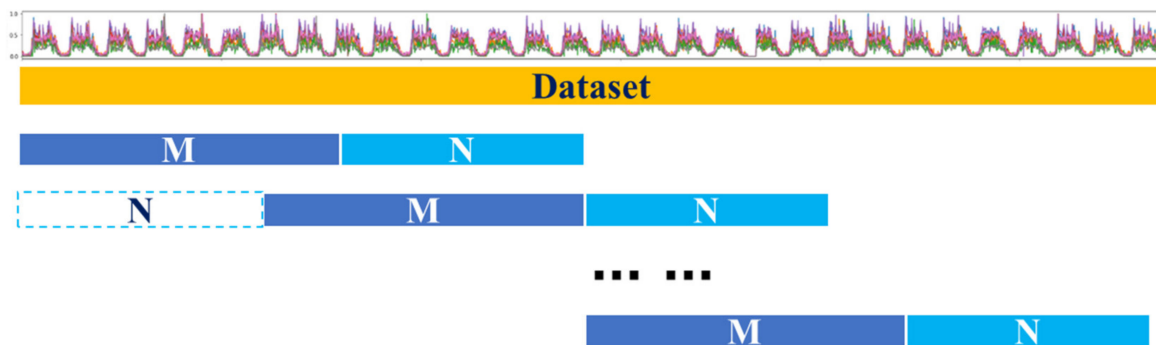


Figure 3. Rolling time-domain training and testing.

Traditional stationary methods, which use transformation, moving average, and differencing processes to eliminate the temporal periodicity and trend out of sequence, are complicated to be precisely captured and scaled back. Hence, a multi-view fusion method fuses the recent l -minute interval view and another long-term view. As analysis of Figure 1 shows, traffic flows are more or less influenced by closeness, periodicity, and trend. Inspired by these observations, we propose a parametric-vector-based fusion method to fuse three components (i.e., periodicity, closeness, and trend). As different temporal sequences may have different periodicities and trends, the parametric-vector-based fusion method is employed to fuse the multiple temporal views between O_{cycle} (cycle: daily, weekly, biweekly, monthly) and O_{recent} .

$$O = \omega_{recent} \odot O_{recent} + \sum_{cycle} (\omega_{cycle} \odot O_{cycle}) \quad (4)$$

where the learnable parameters, ω_{recent} and ω_{cycle} , adjust the impacts caused by closeness, daily periodicity, weekly periodicity, biweekly periodicity, and monthly trend, and \odot is element-wise multiplication. For each of the different temporal views, a series of flow vectors are fetched and concatenated to construct five inputs as follows.

$$O_{recent} = \text{concat}[b_{t-1}, b_{t-2}, \dots, b_{t-l_r}] \in \mathbb{R}^{l_r} \quad (5)$$

$$O_{cycle=daily} = \text{concat}[b_{t-p_d}, b_{t-2p_d}, \dots, b_{t-l_d \cdot p_d}] \in \mathbb{R}^{l_d} \quad (6)$$

$$O_{cycle=weekly} = \text{concat}[b_{t-p_w}, b_{t-2p_w}, \dots, b_{t-l_w \cdot p_w}] \in \mathbb{R}^{l_w} \quad (7)$$

$$O_{cycle=biweekly} = \text{concat}[b_{t-p_b}, b_{t-2p_b}, \dots, b_{t-l_b \cdot p_b}] \in \mathbb{R}^{l_b} \quad (8)$$

$$O_{cycle=monthly} = \text{concat}[b_{t-p_m}, b_{t-2p_m}, \dots, b_{t-l_m \cdot p_m}] \in \mathbb{R}^{l_m} \quad (9)$$

where l_r , l_d , l_w , l_b , and l_m are the input lengths of recent, daily, weekly, biweekly, and monthly, respectively; p_d and p_w are daily and weekly periods; and p_b and p_m are biweekly and monthly trend spans. Then, IGAN-TF can identify multiple kinds of temporal characteristics. In our approach, the complexity of the input data between O_{recent} and O_{cycle} is

$(l_r + l_d + l_w + l_b + l_m)$, which is much less than the period considered by recurrent neural network-based methods. As time goes by, the periodicity and trend roll along with the recent closeness.

Additionally, BiLSTM with timesteps and long-term memory units can also store valuable temporal information to find the periodicity and trend of flows in the time domain.

3.3.2. To Describe the Local Trend Raised by Special Cases with Steady Accuracy

(b) To describe the local trend raised by special cases with steady accuracy

To enable a steady local trend for IGAN-TF, key indicators are selected to map the multi-view fusion local trend, groups of statistic local information from every five temporal neighbors. In this paper, five indicators are chosen, as shown in Table 2, representing the fluctuation direction and range of the local trend in a stable way. The median is used to express the central tendency of flows regardless of maximum and minimum. Maximum and minimum are indicated for the volatility, while moving average is utilized to embody different kinds of the steady local trend to be fitted. Relative strength index is to introduce the average increment level of fluctuation. To this end, the traffic fluctuation direction and range can be resolved in a reasonable domain. The complex and nonlinear relationship between the targeted traffic flows and these indicators is resolved by the generator and the discriminative model of IGAN-TF. It is worth mentioning that signal phase time is not calculated by multi-view fusion as with other indicators. It is accumulated from green time in the current phases of the lane during every interval, regarding it as an indicator for signal influence on the number of passing flows in urban intersections. Thus, multi-source data, including traffic flows and signal timings, are introduced into the same framework.

Table 2. Multi-dimensional indicators for prediction.

Indicators
Median
Maximum
Minimum
Moving average
Signal phase time

The activation functions, such as Sigmoid function and Tanh function, quickly propagate the error back. The ReLU function stops some neurons from ever being activated, with the gradient being zero. The ELU activation function is introduced, which combines the advantages of Sigmoid and ReLU with a linear zone to mitigate the gradient disappearance and a soft saturation zone to improve the robustness. Finally, the training process, which uses RAdam, batch normalization, and minibatch, can avoid the challenges of the optimal local solution, overfitting, and unsteady accuracy.

3.3.3. To Maintain a Continuous Self-Learning Ability in the Time Domain

(c) To maintain a continuous self-learning ability in the time domain

Multi-view temporal fusion is applied to learn the closeness, periodicity, and trend in the time domain. Different kinds of statistic indicators are fed to learn other properties of the fluctuation of traffic flows, including phase time, which affects the self-learning ability to adapt to the variation of signal timing. Rather than fitting the sequence with a fixed expression or using the latest small dataset for online prediction, a discriminative model is introduced to continuously judge the whole sequence during the online prediction process. Rolling time-domain training promotes the continuous self-learning ability of IGAN-TF by selecting appropriate periodicity and trend properties and satisfying online prediction demand upon the accuracy and calculating speed. The training process, which uses RAdam, batch normalization, and minibatch, can adaptively control the gradients to

achieve optimal solutions by fine-tuning input distribution and parameters during training, accelerating convergence for online applications.

4. Data Source and Analysis

Figure 4 depicts the intersection of Hongzehu Road and Qingnian Road in Suqian City, Jiangsu Province, China. The traffic flow and signal timings dataset were stochastically chosen from this intersection. The interval length of observed traffic flow was 15 min, and the time-domain was from 26 October 2016 to 9 May 2017 (a total of 18,816 intervals of 196 days). For a neural network, necessary data analysis is essential [29,30]. We conducted a series of data analyses to show the different patterns of lanes.

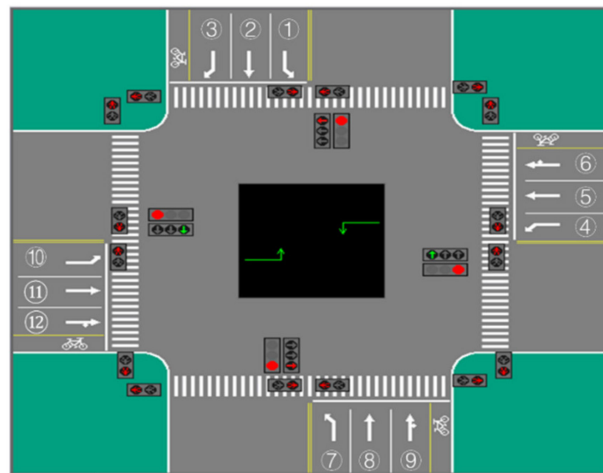


Figure 4. Signalized intersection layout with a group of lanes.

In Figure 4, there are 12 lanes in the intersection. The Talib library was used to calculate the indicators for every four neighboring intervals (representing one-hour intervals). The flow distribution probability of different lanes is shown in Figure 5. Distinct lanes presented different traffic flow data in 196 days; lanes 1, 4, 7, and 10 were left-lane, lanes 2, 5, 8, and 11 were straight-lane, lane 3 was right-lane, and lanes 6, 9, and 12 were straight right-lane. The flow in the east-west direction was much greater than that in the north-south direction because the east-west direction is the main lane for people to commute and tide happening.

The augmented Dickey–Fuller (ADF) test and Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test were adopted to verify the time stability of traffic flow data. The inspection chart is shown in Figure 6. The ADF test, one of the most common statistical tests, can be adapted to determine the unit roots' existence in a sequence and thus help to determine whether the sequence is stationary. It can be seen from Table 3 that the test statistics of lane 1 to lane 12 were less than the critical value, so the original hypothesis was rejected. That is to say, the linear stationarity and differential stationarity of the time series were excellent. The KPSS test results show that the test statistics of lane 1 to lane 12 were less than the critical value, so the original hypothesis was rejected. That is, the trend stability of the data time series was good.

The impact of signal control is very vital to traffic flow. Thus, as in Figure 2, we put signal timing phase time into our model as an indicator. As shown in Figure 7, signal timing phase times of different lanes in a whole day were calculated by each 15-min interval and used to help prediction.

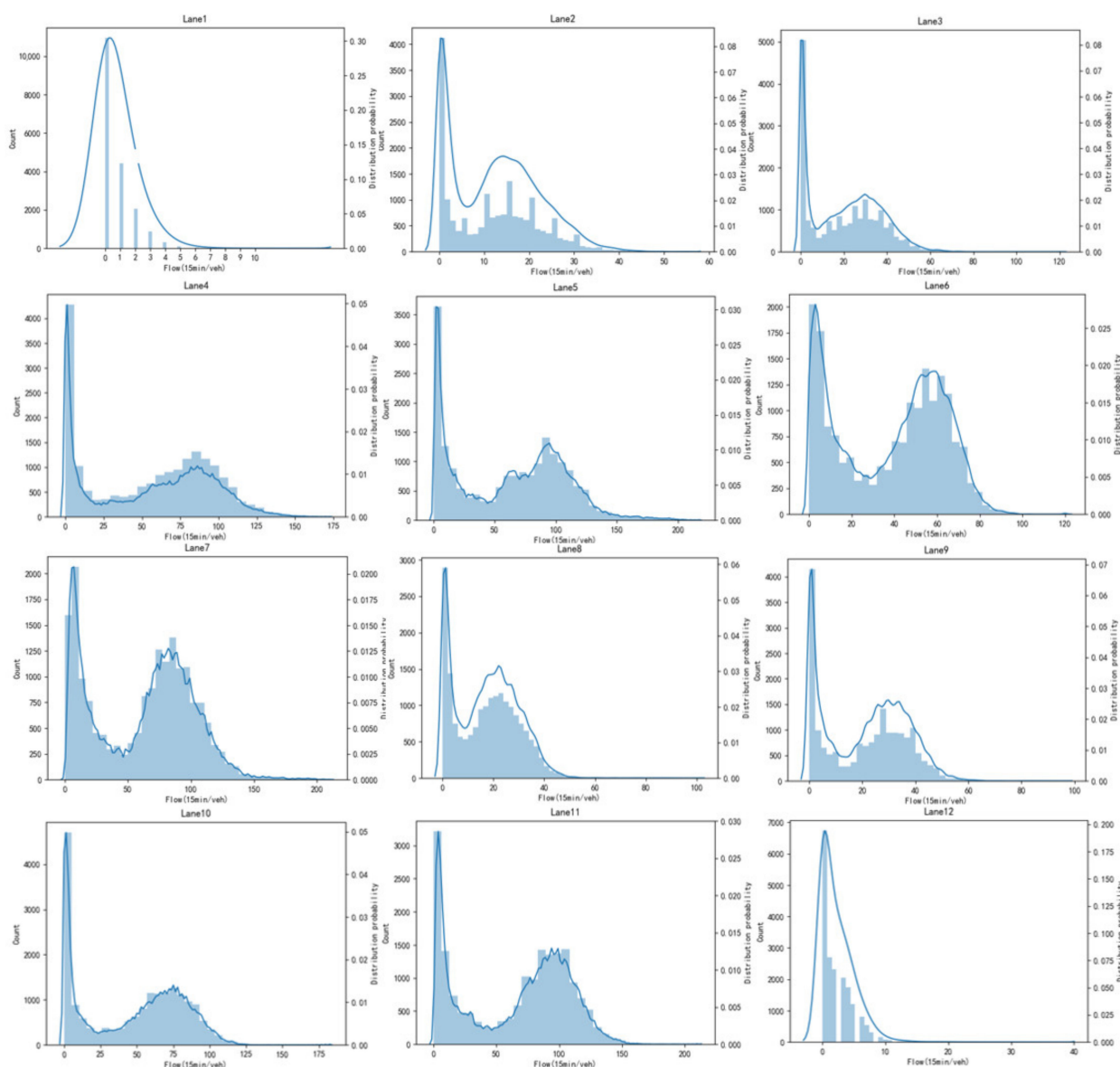


Figure 5. Distribution probability of different lanes.

Table 3. Result of time series stationary test.

Test Results of ADF		Test Results of KPSS	
Critical value (1%)	−3.43249	Critical value (10%)	0.34700
Critical value (5%)	−2.86248	Critical value (5%)	0.46300
Critical value (10%)	−2.56727	Critical value (2.5%)	0.57400
Text statistic of lane 1	−10.21932	Critical value (1%)	0.73900
Text statistic of lane 2	−12.95492	Text statistic of lane 1	0.04918
Text statistic of lane 3	−13.26482	Text statistic of lane 2	0.06788
Text statistic of lane 4	−13.03892	Text statistic of lane 3	0.00759
Text statistic of lane 5	−12.44658	Text statistic of lane 4	0.01037
Text statistic of lane 6	−13.84145	Text statistic of lane 5	0.01035
Text statistic of lane 7	−14.07562	Text statistic of lane 6	0.02243
Text statistic of lane 8	−13.01259	Text statistic of lane 7	0.01891
Text statistic of lane 9	−14.90140	Text statistic of lane 8	0.04274
Text statistic of lane 10	−13.96469	Text statistic of lane 9	0.03465
Text statistic of lane 11	−13.93586	Text statistic of lane 10	0.01226
Text statistic of lane 12	−11.22403	Text statistic of lane 11	0.01486
		Text statistic of lane 12	0.02474

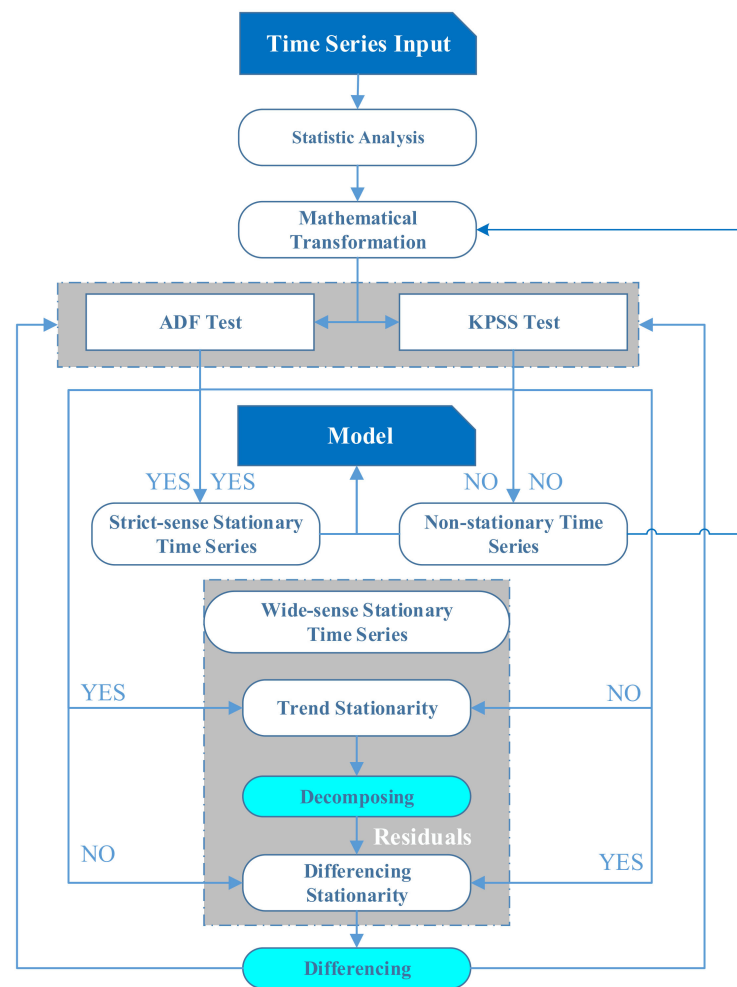


Figure 6. Time series stationary test process.

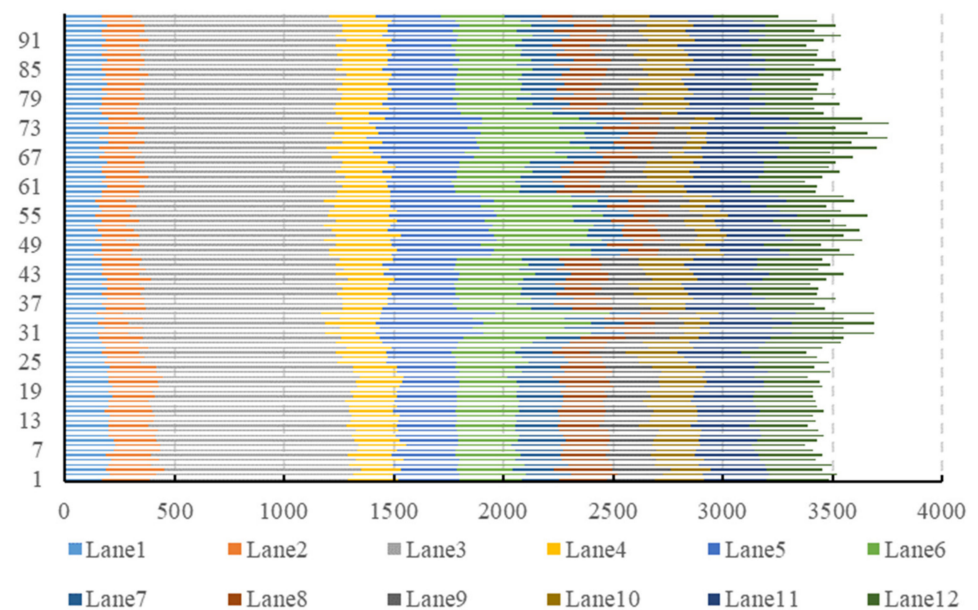


Figure 7. Signal timing phase time of different lanes in a whole day.

5. Discussion

In this section, we use the ground truth traffic flow datasets for the experiment to evaluate the IGAN-TF. Comparative experiments and effective analysis of different components were conducted on the state-of-the-art benchmark deep learning models to demonstrate the predicting performance of the IGAN-TF framework.

5.1. Hardware Environment and Implement Settings

Here are the experimental details and parameter settings of these models:

1. Hardware environment

The Python library Keras based on TensorFlow was adopted to construct our models. A PC Server completed all experiments (the configuration was an Intel Xeon CPU E5-1650, 3.50 GHz, 64GB memory). In this paper, the dataset of every two months was applied to figure out multi-view fusion for calculating the indicators of the generator's input data, except signal phase time in the same view. The fused indicators were trained in a rolling scheme. M was set to one week and N was set to one day. Under the rolling time-domain training scheme, the evolution process of the online prediction was recycled every day since N was set to one day. The computing time for predicting each day was 1.5 h. Considering the mediocre computer utilized and evening hours available for training, IGAN-TF is realizable for online applications.

2. Network Architecture

The implementation details of the IGAN-TF framework are shown in Tables 4 and 5 for the architecture of the generator and the discriminative model, respectively. The parameters of BiLSTM were initialized based on the normal distribution $N(0, 1)$. Learning rates of the generator and the discriminative model were $\rho_G = 0.001$ and $\rho_D = 0.02$. To avoid overfitting with a ratio of 20%, the early stopping and dropout layers were used. δ as the Huber loss function threshold was defined as 1. In RAdam, exponential decay moving 1st moment, m_0 , and exponential decay moving 2nd moment, v_0 , were set to zero. Decay rates, β_1 and β_2 , were set to 0.9 and 0.999, respectively.

Table 4. Network architecture of generator.

Layer (Type)	Output Shape
Input Layer	(None, 96, 8)
LSTM	(None, 96, 32)
batch_normalization_1	(None, 96, 32)
Dropout1	(None, 96, 32)
Bidirection1	(None, 96, 128)
batch_normalization_2	(None, 96, 128)
Dropout2	(None, 96, 128)
Bidirection2	(None, 96, 256)
batch_normalization_3	(None, 96, 256)
Dropout3	(None, 96, 256)
Bidirection3	(None, 512)
batch_normalization_4	(None, 512)
Dropout4	(None, 512)

Table 5. Network architecture of discriminative model.

Layer (Type)	Output Shape
Convolution 1	Filter $64 \times 2 \times 1$, strides 1, ELU
MaxPooling1	2
Convolution 2	Filter $128 \times 2 \times 1$, strides 1, ELU
MaxPooling2	2
Dense3	50, ELu
Dense4	1, Sigmoid

The batch_size as a hyperparameter was initially optimized via cross-validation. As Figure 8 shows, by adjusting the size of batches (generally an exponential multiple of two), the efficiency and accuracy of calculation can be improved. At the same time, the number of iterations required to run the full dataset epoch was reduced. Too large a batch_size is time-consuming for more epochs and may lead to a locally optimal solution, while a small one may lead to the difficulty of convergence. Based on the cross-validation performance of the dataset of different lanes, the best batch_size of lanes 1, 2, 3, 5, 6, 7, 8, 9, 11, and 12 was 16, and that of lanes 4 and 10 was 32.

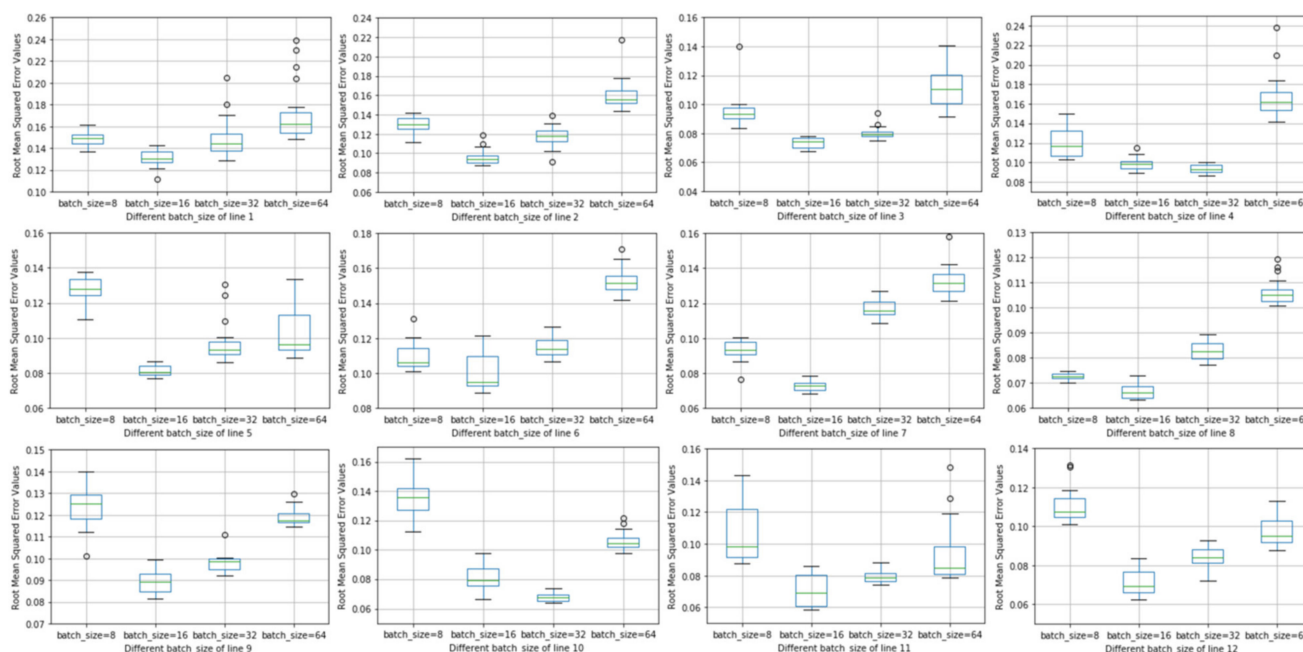


Figure 8. Comparison of different batch_size.

Figure 9, in ascending order of lane number, shows the adversarial loss of 12 lanes based on IGAN-TF. Most of the lanes' results converged after 500 and 700 epochs, with adversarial loss fluctuating around 0.5. The training process was more accurate, steadier, and faster than that of GAN-TF, meaning that the calculating speed to convergence was highly improved by RAdam and other strategies, such as loss choice and multi-view temporal fusion.

5.2. Benchmark Methods

IGAN-TF was compared with the following four benchmarks:

1. ARIMA: Autoregressive integrated moving average is mainly used in time series analysis, such as in the analysis of traffic flows, whose data have non-stationarity. To eliminate the non-stationarity, ARIMA can use an initial differencing step one or more times.
2. BiLSTM: Bidirectional long short-term memory networks use a finite sequence to estimate each element of traffic flows based on the past and future flows, which is done by concatenating the outputs of two LSTM networks, one processing the flow sequence from left to right, the other one from right to left. The input data (indicators and multiple temporal views) and optimizer RAdam of BiLSTM are the same as for IGAN-TF. The loss of BiLSTM is the Huber loss.
3. GAN-TF: Compared with IGAN-TF, generative adversarial networks for traffic flow utilize indicators of the median, maximum, minimum, moving average, and signal timing phase time in neighbor intervals without multi-view temporal fusion in the data preparation aspect. The loss of generator is the Huber loss, and the training optimizer uses RAdam.

4. GAN-NST: Compared with GAN-TF and IGAN-TF, generative adversarial networks for non-signal traffic introduce multi-view temporal fusion and the Huber loss of the generator. RAdam is applied to resolve the training process, and the parameters are mostly set to the same values as those of IGAN-TF. The main difference is that GAN-NST utilizes indicators without signal timing phase time.

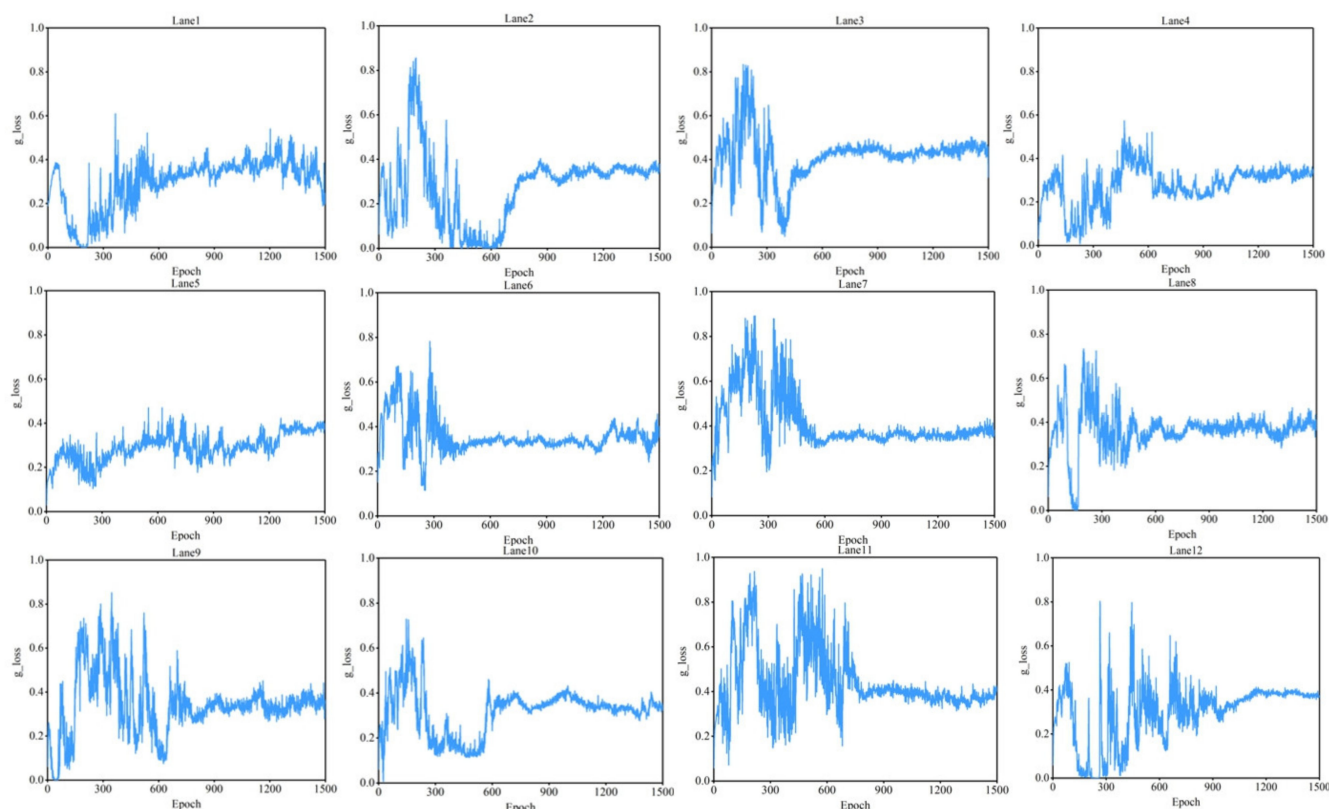


Figure 9. GAN-TF predicted loss of each lane flow.

Based on comparison results of the above benchmarks with IGAN-TF, ARIMA represents the prediction performance of traditional nonlinear methods; BiLSTM with the same input and optimizer denotes the online feedback effect of the discriminative model; GAN-TF shows the contribution of stability and accuracy from multi-source indicators; and GAN-NST presents the function of considering impacts of signal phase time on prediction.

Figure 10 presents a comparison between the predicted and ground-truth traffic flows of each lane by different GAN-based methods for 15 May 2017, based on traffic flows from 8 May 2017 to 14 May 2017. GAN-NST performed the worst in general because it produced many GAN outliers in each lane, showing unsteady performance based on non-signal data as an indicator. It is apparent that GAN-TF and GAN-NST are worse than IGAN-TF. The reason for the tiny difference between GAN-TF and IGAN-TF is that only several outliers were produced. More specific details should be derived from the tables of the average evaluation performance. As the fusion of IGAN-TF is temporal multi-view, the reason why IGAN-TF outperformed GAN-TF might lie in the advantage of the discriminative model.

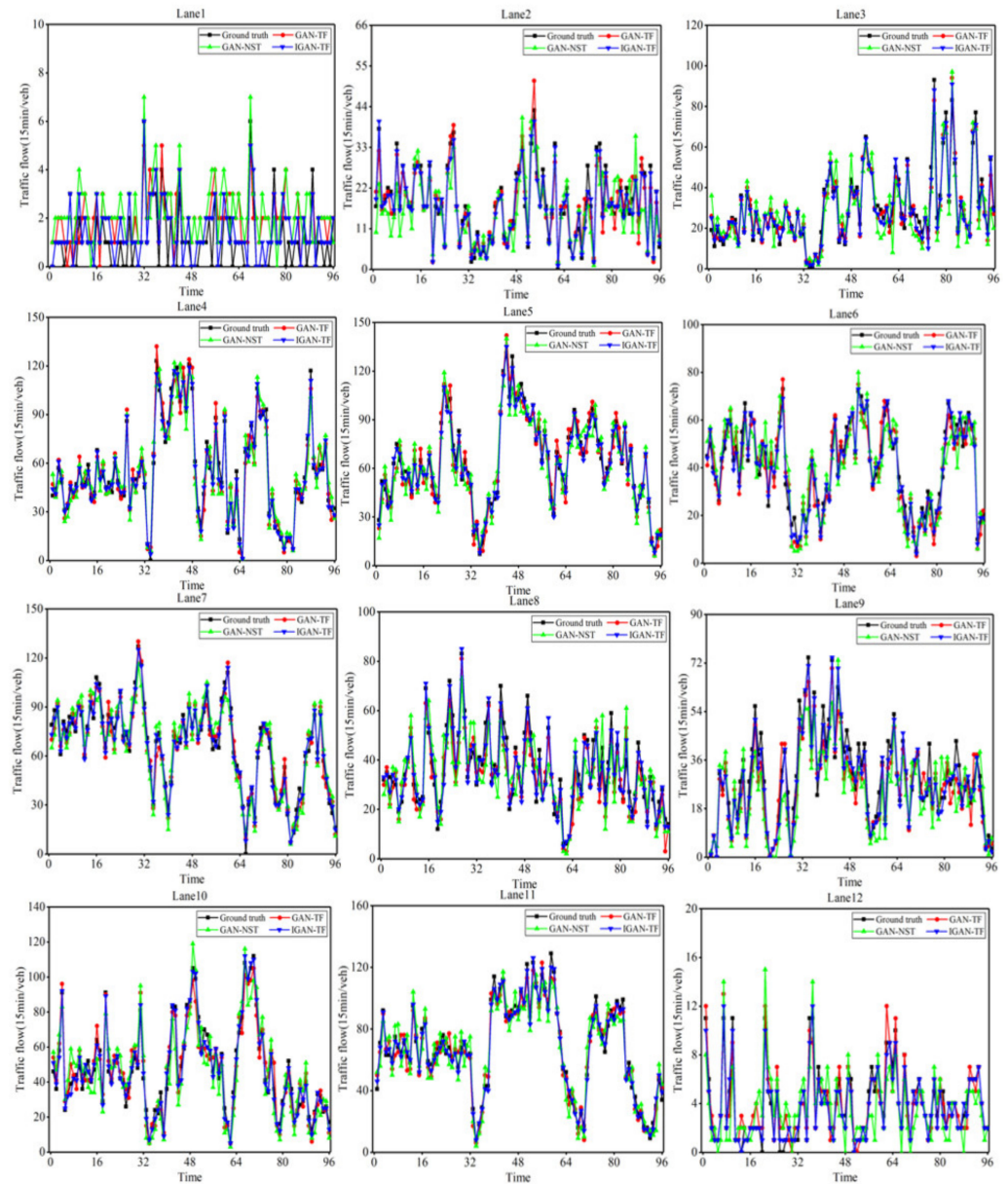


Figure 10. Comparison of predicted flows by models and ground truth of each lane.

5.3. Evaluation Index

To evaluate different prediction methods, we employed root mean square error (RMSE) as the evaluation index. Given ground truth values f_t and predicted values \hat{f}_t , the RMSE was determined as below:

$$RMSE = \sqrt{\frac{1}{T} \sum_{t=1}^T (f_t - \hat{f}_t)^2} \quad (10)$$

5.4. Comprehensive Results

Without multi-view temporal fusion and different indicators, GAN-TF shows worse performance than IGAN-TF. In addition, GAN-NST was worse than BiLSTM and IGAN-TF because it ignored the influence of signal timing phase time as multi-source input data with traffic flows. RMSEs by ARIMA, BiLSTM, GAN-TF, GAN-NST, and IGAN-TF of all lanes in all days are shown in Table 6. Except for lanes 3 and 12, the average RMSE of IGAN-TF is obviously superior to ARIMA, BiLSTM, GAN-TF, and GAN-NST. If no signal timing information is added, only the RMSEs of lanes 3, 4, 6, 7, 8, 9, and 11 are better than

ARIMA and BiLSTM, indicating the importance of traffic signal timing information for traffic prediction. In addition, BiLSTM performed better than ARIMA in most cases. In the data of lane 1, the proportion of zero was as high as 58.18%. The prediction error of lane 1 was relatively large because of the serious fluctuation, many extreme values like zero values, and low flows. Lanes 6, 9, and 12 are straight right-lanes, so there was a lot of mixed traffic flow, including straight and right. The prediction accuracy of these three lanes was lower than that of other lanes.

Table 6. RMSE of all lanes predicted throughout the day.

Lane	1	2	3	4	5	6	7	8	9	10	11	12
ARIMA	0.190	0.193	0.079	0.119	0.096	0.224	0.294	0.194	0.249	0.131	0.179	0.116
BiLSTM	0.179	0.131	0.075	0.107	0.084	0.190	0.158	0.182	0.217	0.097	0.134	0.083
GAN-TF	0.141	0.127	0.067	0.087	0.080	0.162	0.090	0.118	0.162	0.093	0.089	0.098
GAN-NST	0.182	0.133	0.072	0.096	0.092	0.179	0.125	0.159	0.185	0.126	0.101	0.113
IGAN-TF	0.128	0.095	0.071	0.086	0.080	0.115	0.075	0.067	0.119	0.075	0.076	0.096

Performance comparisons between IGAN-TF and other baseline methods in peak hours (7–9 am and 5–7 pm) are presented in Tables 7 and 8.

Table 7. RMSE of morning peak of each lane.

Lane	1	2	3	4	5	6	7	8	9	10	11	12
ARIMA	0.103	0.144	0.067	0.100	0.080	0.200	0.164	0.085	0.221	0.101	0.100	0.067
BiLSTM	0.125	0.101	0.070	0.092	0.067	0.189	0.070	0.093	0.216	0.099	0.089	0.053
GAN-TF	0.098	0.073	0.067	0.080	0.049	0.141	0.070	0.058	0.181	0.090	0.055	0.073
GAN-NST	0.118	0.091	0.079	0.089	0.071	0.162	0.092	0.082	0.203	0.124	0.077	0.096
IGAN-TF	0.081	0.065	0.061	0.069	0.058	0.089	0.059	0.041	0.105	0.078	0.052	0.049

Table 8. RMSE of evening peak of each lane.

Lane	1	2	3	4	5	6	7	8	9	10	11	12
ARIMA	0.118	0.133	0.068	0.091	0.066	0.190	0.141	0.170	0.222	0.135	0.102	0.052
BiLSTM	0.161	0.091	0.060	0.081	0.051	0.190	0.062	0.133	0.201	0.093	0.089	0.040
GAN-TF	0.138	0.074	0.054	0.070	0.040	0.145	0.061	0.071	0.157	0.090	0.044	0.056
GAN-NST	0.158	0.104	0.081	0.082	0.065	0.179	0.105	0.112	0.190	0.118	0.078	0.071
IGAN-TF	0.093	0.069	0.047	0.063	0.039	0.098	0.054	0.069	0.111	0.078	0.042	0.049

From Tables 7 and 8, comparing the average RMSE of results in the morning peak and evening peak with the results throughout the day, the predicted results of IGAN-TF in the morning peak showed higher accuracy than that throughout the day. The traffic flow of lane 3 predicted by IGAN-TF was predicted more accurately than that by GAN-TF. Meanwhile, that of lane 12 predicted by IGAN-TF was also improved obviously. This is mainly because the extreme values, such as zero, reduce the accuracy of the nonpeak period. Moreover, the volumes of traffic flows are shallow, which might strengthen the time series volatility. As a result, many researchers tend to forecast traffic flows without data collected from 0:00 to 6:00. More specifically, compared with the results throughout the day, the proposed IGAN-TF reduced the average RMSE of 12 lanes by 25.55% in the morning peak and 25.09% in the evening peak. Although lane 12 became better, the prediction in the evening peak of IGAN-TF was still worse than that of GAN-TF. In Figure 10, the true values of traffic flows of lane 12 presented apparent high volatility.

Additionally, lane 12 is a straight-right lane, which is different from lane 1 and lane 2. The proportion of right-turn vehicles may variate heavily, which influences the efficiency of signal timing indicators and enables more vehicles to pass the lane without signal control. IGAN-TF has been proven to apply to the prediction traffic flows. Additionally, the weak

performance of lane 12 and the increase of accuracy in peak hours and straight-lanes illustrate that applying IGAN-TF can reach a better prediction performance in predicting traffic flows of high and stable volumes.

5.5. Effects of Different Components

Different time-domain components more or less influence traffic flows. Different temporal sequences may have different periodicities and trends. Multiple temporal views and different predicting steps were employed to find out the effects of traffic flow.

5.5.1. Multiple Temporal Views in Fusion

We compared the results of different experiments with multiple temporal views for the test content, including by month, bi-week, week, and day. As shown in Figure 11, if traffic flow was predicted with single-day traffic data, the results were not very good. As the temporal view span increased, the results were significantly improved. It can be seen that the RMSE had different amplitude reductions in the 12 lanes of an intersection, indicating that the periodicity and similarity of the time series are essential features of the traffic flow pattern.

We formed an input sequence for the signal timing by randomly adding a fixed effective green time for each set of input information. The temporal information input every day is the same, and the periodicity of the time series can be maintained. In contrast, it was found that the model with the use of fixed sequence input (for example, signal timing information is added to the front of the daily flow data) was slightly better than the model with non-signal timing, but the difference was not large. Using random sequence input (that is, random insertion of timing information), the effect was greatly improved by the flow data of one day.

5.5.2. Different Predicting Steps

Under further analysis, we conducted different time intervals for traffic flow data. Five (0.25 h, 0.5, 1 h, 2 h, and 3 h) different intervals were used as the target sequence, as shown in Figure 12, to train the model. As the time interval increased, the RMSE indicator increased slightly in 12 lanes. All lanes were optimal for the IGAN-TF for changing time intervals. The performance of BiLSTM was also worthy of attention. Although most lanes were not optimal, its stability was not bad. ARIMA became more unstable as time intervals increased. The improvement of overall accuracy and stability after the model improvement was obvious, indicating that IGAN-TF is exceptionally suitable for traffic flow forecasting.

5.5.3. Different Loss Selection

To test the accuracy of long-term prediction of IGAN-TF and benchmarks, we compared the RMSE indices of the 12 lanes across the intersection at different loss functions. Table 9 shows that the Huber loss had a greater effect on traffic flow time series prediction than other losses and was more robust to outliers.

Table 9. Loss comparison of RMSE.

Loss	GAN-TF
Binary_crossentropy	0.124
Mean Squared Error	0.107
Mean Absolute Error	0.116
Huber	0.090

5.6. Effects of Different Intersections

Figure 13 gives the real continuous three intersections of Hongzehu Road and Qingnian Road shown in Figure 13a that is the same as Figure 4, Huanghe and Hongzehu Road shown in Figure 13b that includes four left-lanes, six straight-lanes, two right-lanes, and two straight right-lanes, and Hongzehu Road and Xingfu Road shown in Figure 13c

that includes four left-lanes, six straight-lanes, and four straight right-lanes in Suqian City, Jiangsu Province, China. According to the result of RMSE by using the IGAN-TF predicted model, it can be seen that the overall prediction accuracy was excellent in Figure 14, and the lane difference of the three intersections was not particularly large. Due to many right-lanes on Huanghe and Hongzehu Road, the corresponding RMSE performance was negligible. The other two intersections are straight-right lanes, and the error result of RMSE was more significant. Moreover, the right lanes of Hongzehu Road and Xingfu Road contain the straight direction, which led to a greater RMSE in mixed traffic flow mode. The average accuracy of the three intersections was 0.088, which fully shows the scalability and usability of the proposed improved generative adversarial network framework for traffic flow.

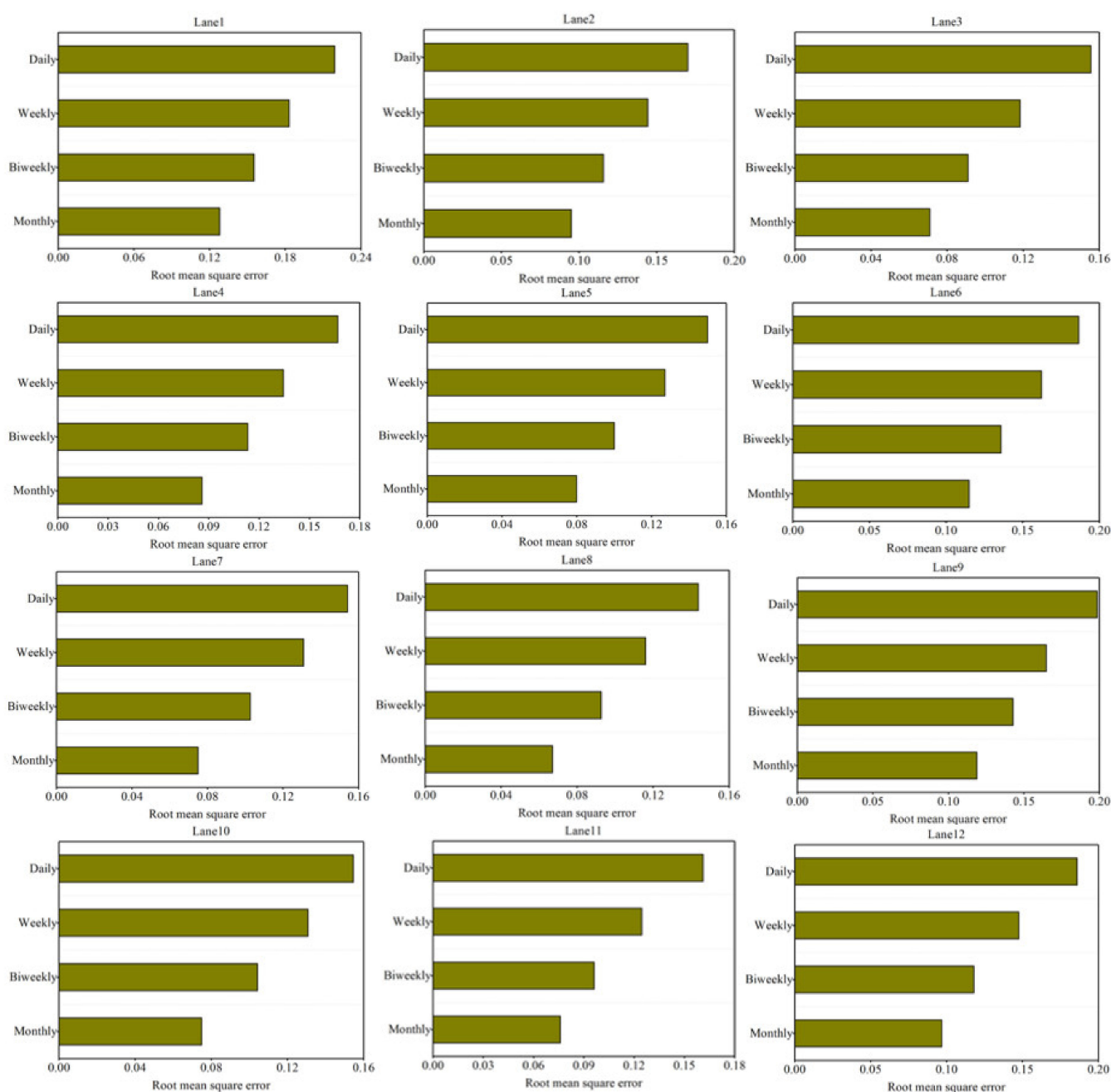


Figure 11. RMSE (percentage) of different groups of temporal views.

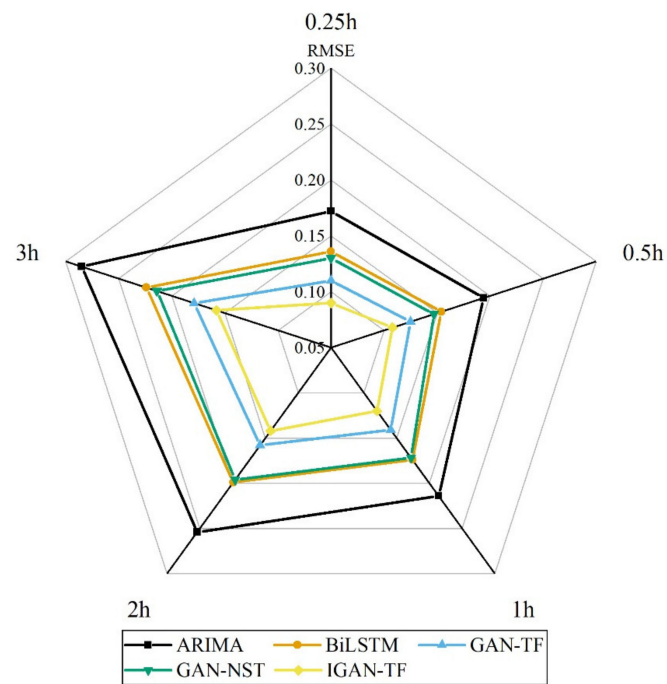


Figure 12. Different predicting steps.

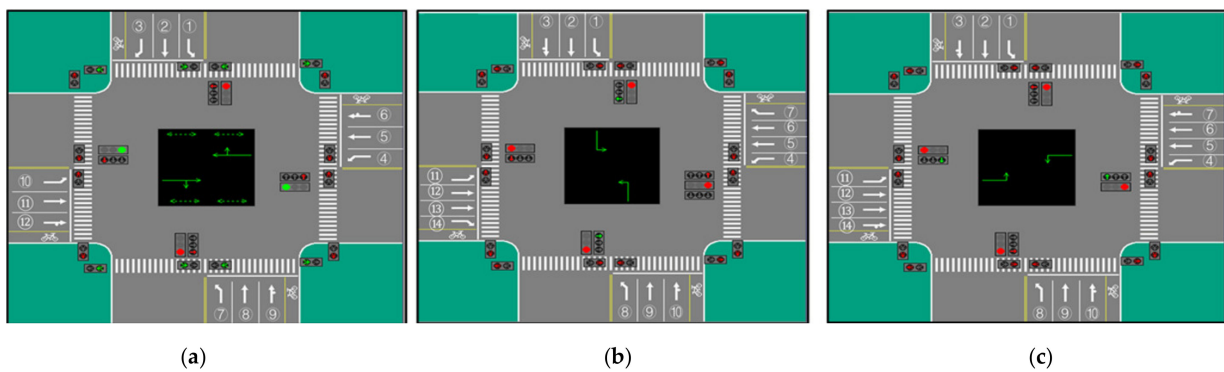


Figure 13. Case of three signalized intersections.

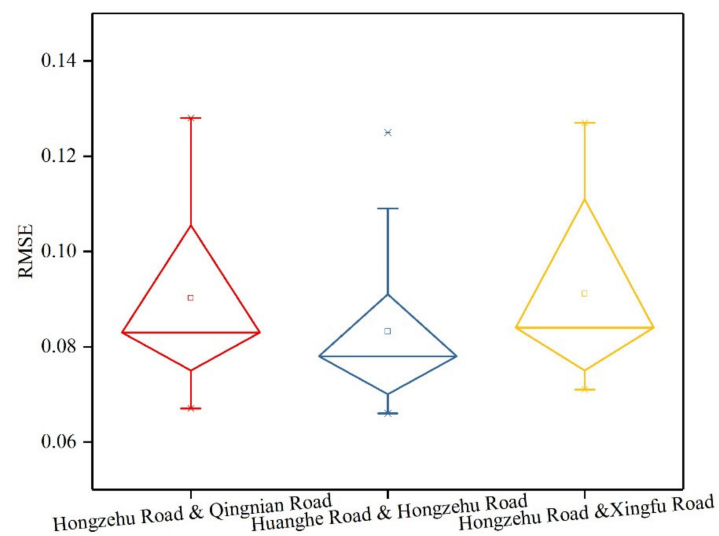


Figure 14. RMSE of three different intersections.

6. Conclusions

Based on the deep learning model LSTM and CNN, this paper proposes IGAN-TF, a GAN-TF framework with generator BiLSTM and discriminant CNN to predict traffic flows in lanes during fixed intervals. The dataset combines indicators of current green time aggregated from phases in signal timings. The average, minimum, and maximum values of neighbor intervals were applied to generate the input of discriminant CNN to compete with the true value flow sequence. Different loss functions were defined to train and test IGAN-TF with an SGD algorithm and a rolling time-domain strategy. We measured the prediction results of IGAN-TF with criteria and compared it with ARIMA and BiLSTM. The morning and evening peak periods and left-turn, straight-forward, and right-turn directions showed the accuracy and stability of IGAN-TF in different time and traffic conditions and its potential to be discovered.

The proposed IGAN-TF and its application suggest a new way to resolve traffic prediction issues. IGAN-TF provides an open framework to combine the advantages of different sub-models and offers a suitable platform for them to compete with each other. As it was initially designed for the machine learning environment, big data scenarios (e.g., traffic prediction and traffic state estimation) can be solved in IGAN-TF. However, it is worth studying the application efficiency of IGAN-TF in various data datasets, distributions, and statistical properties. The indicators and their input methods, which influence the data variation most, significantly improve IGAN-TF. As IGAN-TF has a flexible framework, the cooperation of different approaches and strategies are also an efficient way to optimize the accuracy of IGAN-TF.

Author Contributions: Conceptualization, T.S. and B.S.; methodology, T.S. and B.S.; validation, B.S. and R.H.; writing—original draft preparation, T.S. and B.S.; writing—review and editing, Z.J. and J.X.; supervision, T.S.; funding acquisition, T.S. and R.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Key Research and Development Program of China (2018YFB1601000), National Natural Science Foundation of China (52131204) and Shanghai Science and Technology Innovation Action Plan Project (19DZ1209004).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sun, B.; Sun, T.; Zhang, Y.; Jiao, P. Urban traffic flow online prediction based on multi-component attention mechanism. *IET Intell. Transp. Syst.* **2020**, *14*, 1249–1258. [\[CrossRef\]](#)
2. Xie, D.F.; Fang, Z.Z.; Jia, B.; He, Z. A data-driven lane-changing model based on deep learning. *Transp. Res. Part C Emerg. Technol.* **2019**, *106*, 41–60. [\[CrossRef\]](#)
3. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *arXiv* **2014**, arXiv:1406.2661.
4. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv* **2015**, arXiv:1511.06434.
5. Mescheder, L.; Nowozin, S.; Geiger, A. The numerics of gans. *arXiv* **2017**, arXiv:1705.10461.
6. Mescheder, L.; Geiger, A.; Nowozin, S. Which training methods for GANs do actually converge? In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 3481–3490.
7. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein generative adversarial networks. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 214–223.
8. Zhang, K.; Jia, N.; Zheng, L.; Liu, Z. A novel generative adversarial network for estimation of trip travel time distribution with trajectory data. *Transp. Res. Part C Emerg. Technol.* **2019**, *108*, 223–244. [\[CrossRef\]](#)
9. Van Der Voort, M.; Dougherty, M.; Watson, S. Combining Kohonen maps with ARIMA time series models to forecast traffic flow. *Transp. Res. Part C Emerg. Technol.* **1996**, *4*, 307–318. [\[CrossRef\]](#)

10. Ding, Q.Y.; Wang, X.F.; Zhang, X.Y.; Sun, Z.Q. Forecasting traffic volume with space-time ARIMA model. *Adv. Mater. Res.* **2001**, *156*, 979–983. [\[CrossRef\]](#)
11. Chandra, S.R.; Al-Deek, H. Predictions of freeway traffic speeds and volumes using vector autoregressive models. *J. Intell. Transp. Syst.* **2009**, *13*, 53–72. [\[CrossRef\]](#)
12. Huang, H.Q.; Tang, T.H. Short-term traffic flow forecasting based on ARIMA-ANN. In Proceedings of the IEEE International Conference on Control and Automation, ICCA, Guangzhou, China, 30 May–1 June 2007; pp. 2370–2373.
13. Gong, Y.S.; Zhang, Y. Research of short-term traffic volume prediction based on Kalman filtering. In Proceedings of the 6th International Conference on Intelligent Networks and Intelligent Systems (ICINIS), Shenyang, China, 1–3 November 2013; pp. 99–102.
14. Ojeda, L.L.; Kibangou, A.Y.; De Wit, C.C. Adaptive Kalman filtering for multi-step ahead traffic flow prediction. In Proceedings of the American Control Conference, Washington, DC, USA, 17–19 June 2013; pp. 4724–4729.
15. Guo, J.; Huang, W.; Williams, B.M. Adaptive Kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification. *Transp. Res. Part C Emerg. Technol.* **2014**, *43*, 50–64. [\[CrossRef\]](#)
16. Huang, W.; Song, G.; Hong, H.; Xie, K. Deep architecture for traffic flow prediction: Deep belief networks with multitask learning. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 2191–2201. [\[CrossRef\]](#)
17. Lv, Y.; Duan, Y.; Kang, W.; Li, Z.; Wang, F.Y. Traffic flow prediction with big data: A deep learning approach. *IEEE Trans. Intell. Transp. Syst.* **2014**, *16*, 865–873. [\[CrossRef\]](#)
18. Sun, B.; Sun, T.; Jiao, P. Spatio-Temporal Segmented Traffic Flow Prediction with ANPRS Data Based on Improved XGBoost. *J. Adv. Transp.* **2021**, *2021*, 5559562. [\[CrossRef\]](#)
19. Khotanzad, A.; Sadek, N. Multi-scale high-speed network traffic prediction using combination of neural networks. In Proceedings of the International Joint Conference on Neural Networks, Portland, OR, USA, 20–24 July 2003; Volume 2, pp. 1071–1075.
20. Qiu, C.; Wang, C.; Zuo, X.; Fang, B. A bayesian regularized neural network approach to short-term traffic speed prediction. In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, Istanbul, Turkey, 10–13 October 2010; pp. 2215–2220.
21. Ma, X.; Yu, H.; Wang, Y.; Wang, Y. Large-scale transportation network congestion evolution prediction using deep learning theory. *PLoS ONE* **2015**, *10*, e0119044. [\[CrossRef\]](#) [\[PubMed\]](#)
22. Tan, H.; Xuan, X.; Wu, Y.; Zhong, Z.; Ran, B. A comparison of traffic flow prediction methods based on DBN. In Proceedings of the 16th COTA International Conference of Transportation Professionals (CICTP), Shanghai, China, 6–9 July 2016; pp. 273–283.
23. Cui, Z.; Ke, R.; Pu, Z.; Wang, Y. Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction. *arXiv* **2018**, arXiv:1801.02143.
24. Wang, K.; Gou, C.; Duan, Y.; Lin, Y.; Zheng, X.; Wang, F.Y. Generative adversarial networks: Introduction and outlook. *IEEE/CAA J. Autom. Sin.* **2017**, *4*, 588–598. [\[CrossRef\]](#)
25. Wang, K.; Gou, C.; Wang, F.Y. Parallel vision: An ACP-based approach to intelligent vision computing. *Acta Autom. Sin.* **2016**, *42*, 1490–1500.
26. Liang, Y.; Cui, Z.; Tian, Y.; Chen, H.; Wang, Y. A deep generative adversarial architecture for network-wide spatial-temporal traffic-state estimation. *Transp. Res. Rec.* **2018**, *2672*, 87–105. [\[CrossRef\]](#)
27. Lin, Y.; Dai, X.; Li, L.; Wang, F.Y. Pattern sensitive prediction of traffic flow based on generative adversarial framework. *IEEE Trans. Intell. Transp. Syst.* **2018**, *20*, 2395–2400. [\[CrossRef\]](#)
28. Liu, L.; Jiang, H.; He, P.; Chen, W.; Liu, X.; Gao, J.; Han, J. On the variance of the adaptive learning rate and beyond. *arXiv* **2019**, arXiv:1908.03265.
29. Du, W.; Sun, B.; Kuai, J.; Xie, J.; Yu, J.; Sun, T. Highway travel time prediction of segments based on ANPR data considering traffic diversion. *J. Adv. Transp.* **2021**, *2021*, 9512501. [\[CrossRef\]](#)
30. Izonin, I.; Tkachenko, R.; Fedushko, S.; Koziy, D.; Zub, K.; Vovk, O. RBF-Based Input Doubling Method for Small Medical Data Processing. In *Lecture Notes on Data Engineering and Communications Technologies, Proceedings of the International Conference on Artificial Intelligence and Logistics Engineering, Kyiv, Ukraine, 22–24 January 2021*; Springer: Cham, Switzerland; Volume 82, pp. 23–31.