Graph Convolutional Adversarial Networks for Spatiotemporal Anomaly Detection

Leyan Deng, Defu Lian, Zhenya Huang, and Enhong Chen, *Senior Member, IEEE*

***Abstract*— Traffic anomalies, such as traffic accidents and unexpected crowd gathering, may endanger public safety if not handled timely. Detecting traffic anomalies in their early stage can benefit citizens' quality of life and city planning. However, traffic anomaly detection faces two main challenges. First, it is challenging to model traffic dynamics due to the complex spatiotemporal characteristics of traffic data. Second, the criteria of traffic anomalies may vary with locations and times. In this article, we propose a spatiotemporal graph convolutional adversarial network (STGAN) to address these above challenges. More specifically, we devise a spatiotemporal generator to predict the normal traffic dynamics and a spatiotemporal discriminator to determine whether an input sequence is real or not. There are high correlations between neighboring data points in the spatial and temporal dimensions. Therefore, we propose a recent module and leverage graph convolutional gated recurrent unit (GCGRU) to help the generator and discriminator learn the spatiotemporal features of traffic dynamics and traffic anomalies, respectively. After adversarial training, the generator and discriminator can be used as detectors independently, where the generator models the normal traffic dynamics patterns and the discriminator provides detection criteria varying with spatiotemporal features. We then design a novel anomaly score combining the abilities of two detectors, which considers the misleading of unpredictable traffic dynamics to the discriminator. We evaluate our method on two real-world datasets from New York City and California. The experimental results show that the proposed method detects various traffic anomalies effectively and outperforms the state-of-the-art methods. Furthermore, the devised anomaly score achieves more robust detection performances than the general score.**

***Index Terms*— Adversarial learning, anomaly detection, intelligent transportation, spatiotemporal data mining.**

Manuscript received December 7, 2020; revised June 24, 2021 and November 22, 2021; accepted December 11, 2021. Date of publication January 4, 2022; date of current version June 2, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2020AAA0103800, in part by the National Natural Science Foundation of China under Grant 61976198 and Grant 62022077, and in part by the Fundamental Research Funds for the Central Universities under Grant WK2150110017. *(Corresponding author: Defu Lian.)*

Leyan Deng is with the School of Data Science, University of Science and Technology of China, Hefei 230026, China (e-mail: dleyan@mail.ustc.edu.cn).

Defu Lian and Enhong Chen are with the School of Data Science, University of Science and Technology of China, Hefei 230026, China, and also with the Yangtze River Delta Information Intelligence Innovation Research Institute, Wuhu 241000, China (e-mail: liandefu@ustc.edu.cn; cheneh@ustc.edu.cn).

Zhenya Huang is with the School of Computer Science and Technology, University of Science and Technology of China, Hefei 230026, China (e-mail: huangzhy@ustc.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2021.3136171.

Digital Object Identifier 10.1109/TNNLS.2021.3136171

I. Introduction

THE rapid progress of the social economy has facilitated many people's lives. However, with the increasing number of vehicles and population in cities, citizens and traffic management face many serious traffic problems, such as traffic congestion, traffic accidents, and unexpected crowd gathering [1]. Such traffic anomalies may endanger public safety and stability if not timely handled. For example, on January 26, 2017, in Harbin, a single traffic accident finally caused a serial rear-end collision since the earliest accident was not detected in time [2]. Therefore, one of the fundamental issues for traffic management and public safety is to detect traffic anomalies timely. It has great significance from both manager and citizen perspectives as follows [3].

1) For traffic management, detecting anomalies timely and even predicting anomalies are instrumental in preventing serious accidents from occurring. Specifically, the traffic management department can handle sudden accidents depending on reported locations and improve urban planning to eliminate accident-prone regions [4]–[6].
2) For citizens, detecting and predicting urban anomalies can benefit citizens' quality of life [7]. For instance, detecting traffic anomalies can provide real-time traffic conditions. Therefore, the citizens can adjust the route to reduce the travel time and cost.

Nowadays, large-scale sensors and mobile devices have produced a variety of traffic data, such as taxi trip records and traffic dynamics data [8]–[10]. When an anomaly happens, abnormal changes can be found from these data sources. Therefore, we can detect these anomalies listed above with the help of massive traffic data. However, there are critical challenges in traffic data mining due to the following issues.

1) *Complexity of Traffic Data:* Traffic data are influenced by many complex factors and present dynamic spatiotemporal features. Moreover, the same change that occurred in different locations and times may mean normal or abnormal. For example, it is normal for the traffic flow of the office area to increase in the morning on weekdays. However, when a similar increase occurs on weekends or in the recreation area, it represents abnormal. Therefore, it is challenging and critical to model the complex factors of traffic dynamics and anomalies [4], [11], [12].
2) *Lack of Reported Traffic Anomalies:* The traffic anomalies are usually not recorded or critical information is lacking when recorded. Therefore, most existing works

are unsupervised, which learns latent features from traffic data to represent normal patterns and then regards deviating from normal patterns as abnormal. However, their training goals are to predict or reconstruct, which will lead to suboptimal anomaly detection performance.

3) *Diversity of Traffic Anomalies:* There are various kinds of traffic anomalies in the real world, which presents different patterns in traffic data. For instance, social events such as concerts would cause a sharp increase in traffic flow within several hours [13], [14]. On the contrary, traffic accidents would cause a sharp decrease in traffic speed. In addition, traffic anomalies are affected by many factors. For example, the traffic flow of the road under construction would decrease. However, the adjacent roads may have traffic flow increase since most people change driving paths. Therefore, the criteria to determine traffic anomalies should vary with spatiotemporal features.

To address these issues, we propose a traffic anomaly detection method based on spatiotemporal graph convolutional adversarial network (STGAN). First, to capture the complex spatiotemporal dependencies of traffic data, we propose a spatiotemporal generator to predict the normal traffic dynamics. Specifically, considering the high correlations between neighboring data in spatial and temporal dimensions, the main component of the generator is the recent module, which aims at learning short-term spatiotemporal features. The other components are devised to learn long-term temporal features and external features, respectively. Second, we propose a spatiotemporal discriminator to provide a criterion varying with locations and times. Furthermore, to solve the lack of labeled data in traffic anomaly detection works, the generator generated fake negative data to help the training of the discriminator. Therefore, after adversarial training, both the generator and the discriminator are able to detect traffic anomalies. Finally, we design a novel anomaly score to combine the detection abilities of two components. The main contributions of this article are as follows.

1) We propose an STGAN for traffic anomaly detection. The generator aims to model the normal traffic patterns, consisting of three components to learn spatiotemporal features, historical trend features, and external features. The discriminator determines whether an input sequence is real or fake, that is, normal or abnormal. After adversarial training, the generator and the discriminator can be independently used as detectors.
2) We devise a flexible anomaly score combining the detection performances of the generator and the discriminator. The anomaly score magnifies the difference between the normal and abnormal data by considering the unpredictability of traffic dynamics. The effectiveness of the proposed anomaly score is demonstrated in the experiments.
3) We conduct extensive experiments on the NYC urban flow dataset and California freeway dataset. The experimental results show that the proposed STGAN outperforms three types of baselines, including individual methods, temporal methods, and spatiotemporal methods. We also conduct the ablation experiments and prediction performance comparison.

The rest of this article is organized as follows. We first present the related work regarding traffic anomaly detection and GAN-based methods in Section II. Then, we preprocess and analyze the two real-world datasets in Section III. Some preliminary definitions and an overview of our framework are presented in Section IV. Sections V and VI introduce the spatiotemporal generator and discriminator in detail and then introduce a novel anomaly score. We evaluate our method on two real-world datasets in Section VII and conclude this work in Section VIII.

II. Related Work

With the rapid growth of sensors and mobile devices, there are a variety of traffic data. In recent years, mining traffic data to detect traffic anomalies has attracted extensive attention. In this section, we first introduce the existing works in traffic anomaly detection and then present the anomaly detection methods based on generative adversarial networks.

A. Traffic Anomaly Detection

In recent years, there have been many new works proposed for traffic anomaly detection problems. The existing works can be classified into four main groups, including spatiotemporal feature-based, statistical models, tensor factorization-based, and deep learning models [2].

1) Spatiotemporal Feature-Based Model: Spatiotemporal feature-based methods transform the traffic anomaly detection problem into a classical anomaly detection problem via constructing or learning features from traffic data [3], [15]–[20]. It is hard for constructed features to capture the complex spatial and temporal correlations of traffic data. In order to obtain effective spatiotemporal features, many representation learning methods are proposed. For example, Wang *et al.* [19] detected traffic anomalies using the change of road segments' features. Yang *et al.* [20] introduced a Bayesian RPCA method cofactorizing multiple traffic data streams to learn the latent low-rank and sparse matrices from the traffic data with different measurements. Zhang *et al.* [3] used a similarity-based algorithm to estimate anomaly scores based on the historically similar region and then leveraged a one-class support vector machine (OC-SVM) to give the integrated anomaly score for each region. For spatiotemporal feature-based methods, disjointed anomaly detection and feature extraction may lead to suboptimal detection performance. In this article, we learn an end-to-end anomaly detection discriminator, which learns a flexible anomaly score based on spatiotemporal features.

2) Statistical Model: Statistical models are also commonly used in traffic anomaly detection [21]–[25]. These works model the normal traffic dynamics with a probability distribution and then provide the probability as anomaly scores. For instance, Yang and Liu [24] denoted an observation as a vector representing the number of people in every zone at each time. Then, they applied the K-means clustering technique to group the observation vectors into K clusters. The historical data were used to train hidden Markov model (HMM) to model the observation in each cluster. The anomaly score can be defined as the probability of a new observation sequence. Similarly, Witayangkurn *et al.* [25] clustered the observation vectors and defined anomaly scores for each observation instead of the observation sequence. However, these statistical models do not consider the impact of complex factors on traffic data, such as weather and time. In this article, we propose an independent module to capture the external features.

3) Tensor Factorization-Based Model: Due to the scarcity and complexity of abnormal data, most of the existing works learn normal traffic patterns of traffic data and then regard the data point deviating from the normal patterns as anomalies. In tensor factorization-based methods, the traffic dynamic is considered as a combination of some types of basic patterns. These patterns can be learned by applying restricted tensor factorization techniques on tensor-represented traffic data [26]–[30]. Lin *et al.* [28] applied nonnegative CP decomposition to daily urban flow data to extract basic mobility patterns. During testing, they used basic trained patterns to optimize spatial factor vectors and then applied the local outlier factor (LOF) algorithm to historical spatial factor vectors of each region to detect abnormal regions for one day. Chen *et al.* [29] used the social activity tensor to decompose the crowd movement tensor into several basic patterns. Then, based on multiple outlier detection, they detected anomalies from the basic patterns and used them to detect and describe associated abnormal events. Similarly, Wang *et al.* [30] proposed a neighbor-regularized and context-aware tensor factorization method, which focuses on spatial, temporal, and spatiotemporal patterns. These tensor factorization-based models only learn the spatial features from history traffic data and ignore the high spatial correlations between neighboring regions. In this article, we introduce a recent module to capture the high correlations of spatial neighbors.

4) Deep Learning Model: In recent years, deep neural networks have achieved great success in learning representations of complex data, such as high-dimensional data, temporal data, and spatial data. Some recent works applied deep learning on traffic anomaly detection to model complex traffic dynamics. Trinh *et al.* [31] utilized long short-term memory (LSTM) neural network to capture long-term temporal dependencies from multivariate time series. Then, the anomaly detection problem was addressed as a binary classification problem, and the designed algorithm was used to classify the traffic sequences as normal or abnormal. Trinh *et al.* [31] needed enough abnormal data to train the classification model. However, the labeled data are scarce in the case of traffic anomaly detection. Zhang *et al.* [32] proposed a decomposition method that decomposes urban dynamics into normal and abnormal components. They designed a geoembedding method to learn fixed spatial features. The normal component was learned via a neural network based on the fixed spatial features and handcrafted temporal features. Then, the LOF algorithm was utilized to the abnormal component to achieve the final anomaly score. However, these existing works modeled temporal features only or modeled the spatial and temporal feature separately, which are not able to model the traffic dynamics well.

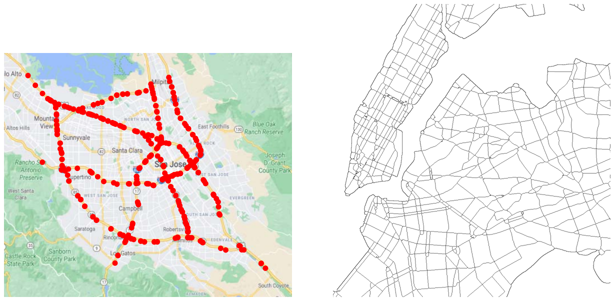
In our work, we capture the dynamic spatiotemporal features of traffic data from multiple data sources. We consider the high correlations of traffic data in the spatial and temporal dimensions. We do not only use the difference between the normal pattern and real traffic dynamic as anomaly score but also propose a discriminator to detect traffic anomalies based on spatiotemporal features. Then, we design a flexible anomaly score for robust detection performance.

B. Generative Adversarial-Based Anomaly Detection

GAN-based [33] models emerged quickly as one of the popular deep anomaly detection approaches after it was proposed [34]. One of the early methods is the AnoGAN [35] for image anomaly detection. The generator generated the realistic instance based on learned latent features of normal instances. The anomaly score was defined by the similarity between the real instance and generated instance. One main limitation of AnoGAN is the computational inefficiency in the iterative search of the latent feature. Akcay *et al.* [36] further improved the generator by changing the generator network to an encoder–decoder–encoder network. The search of the latent feature was formalized as an encoder.

The above methods are all applied in image anomaly detection. Moreover, GAN-based anomaly detection methods are also proposed for video data or time-series data. For instance, Li *et al.* [37] proposed a multivariate time-series anomaly detection framework. This work modeled the complex multivariate correlations by considering the entire variable set concurrently. Then, they devised an anomaly score using discriminant result and reconstruction. Lee *et al.* [38] focused on abnormal detection in videos. They devised a generator based on bidirectional ConvLSTM to generate an interframe by considering spatiotemporal characteristics. The discriminator consisted of 3-D convolutional layers to determine whether the video sequence was real or not. Then, they devised an anomaly score by using the losses of the generator and the discriminator. Ravanbakhsh *et al.* [39] addressed the abnormality detection problem in crowded scenes. They proposed a GAN-based method consisting of two conditional GANs. The local difference between the real and the generated images was considered as an anomaly score at testing time.

In time-series data, we do not need to consider spatial features. In video data, each video frame contains similar spatial features, and there are high correlations between consecutive frames. However, in traffic data, there are dynamic correlations between different locations, and in addition to the correlations between neighboring times, the traffic dynamics present a specific trend in history. In summary, traffic dynamics are affected by many complex factors and have more complex spatiotemporal features than time-series data or video data. Therefore, these GAN-based anomaly detection methods cannot be applied in detecting traffic anomalies directly. In this article, we propose a novel spatiotemporal adversarial network. The proposed generator simultaneously captures the dynamic spatial and temporal feature, and the discriminator detects anomalies at different times and locations.



(a) (b)

Fig. 1. (a) Sensors in the bay area. (b) Segmented regions in New York City.

III. Dataset Analysis

To evaluate the effectiveness of our proposed spatiotemporal anomaly detection method based on adversarial networks, we collect two real-world datasets, namely, urban flow data in New York City and freeway traffic data in California. In this section, we describe the details of two datasets and present preprocessing on the two datasets.

A. Data Collection and Data Preprocessing

1) PeMS Dataset: Caltrans Performance Measurement System (PeMS)[1] provides access to real-time and historical traffic performance on California freeways. The primary data source is the vehicle detector stations (VDSs) distributed on freeway. The coordinates of VDS are also available. We select 365 sensors (VDS) in the bay area visualized in Fig. 1(a). We collect five months of data ranging from January 1, 2014 to May 6, 2014. Each record contains the total volumes and average speeds of the corresponding road segments every 5 min.

We denote a sensor as a node and regard two nodes as adjacent if two sensors are adjacent on the same freeway. We preprocess each record as a vector, representing the respective traffic dynamics of all lanes. Note that the numbers of lanes in different segments are not equal. We set the number of lanes as the maximum value 6 and fill the null value by zero.

2) NYC Dataset: The NYC dataset [1], [40] contains taxicab data[2] and bike renting data[3] in New York City. We collect these two datasets and weather data[4] of NYC from January 15, 2014 to November 30, 2014. Each trip record includes arrival and departure times and locations, and especially, the number of passengers is also available in NYC taxicab data.

We use major roads to partition the entire city, which results in 862 regions shown in Fig. 1(b). We denote each region as a node and consider two nodes adjacent if their boundaries are connected. Then, we divide each day into time slots of 30 min. For each region, we compute the number of leaving trips and arriving trips of taxi and bike during each time slot, respectively. Table I summarizes the statistics of the preprocessed datasets.

[1]http://pems.dot.ca.gov/

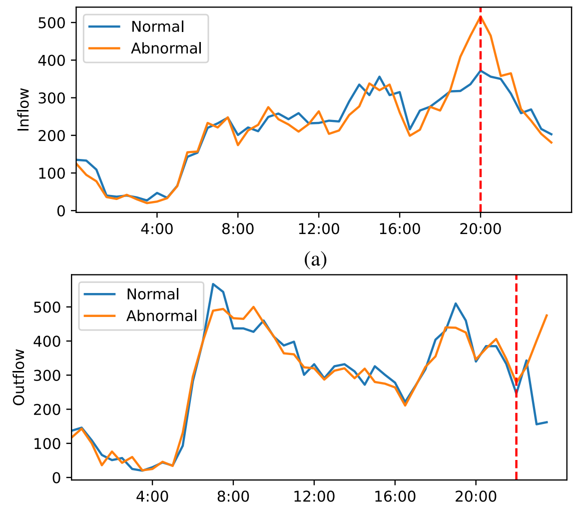
[2]https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page

[3]http://www.citibikenyc.com/system-data

[4]https://tianqi.911cha.com/

TABLE I
Description of Preprocessed Datasets

Properties	NYC Dataset	PeMS Dataset
Number of Nodes	862	365
Time Interval	30 minutes	5 minutes
Length of Feature	4	12
Time Span	1/15/2014 - 11/30/2014	1/1/2017 - 5/6/2017



(a) (b)

Fig. 2. Traffic flows on May 13, 2014 and the same day of last week. (a) Taxi inflow, the dotted line indicates the start time of the concert. (b) Taxi outflow, the dotted line indicates the end time of the concert.

B. Empirical Study

In this section, we conduct empirical studies on two real-world datasets to visualize how different types of traffic anomalies affect the traffic data. Then, we summarize the characteristics of traffic anomalies.

1) Lady Gaga Concert in NYC: On May 13, 2014, Lady Gaga held a concert in Madison Square Garden, which started at 8:00 and ended at 10:00. Fig. 2 shows the taxi flows on that day and the same day last week. As shown in Fig. 2(a), the audiences arrive one after another before the concert and then cause a significant inflow increase compared to last week. Fig. 2(b) shows that the audiences leave this stadium after the concert, presenting opposite traffic dynamics from last week. In summary, social events usually present as sharp changes in traffic flow within several hours.

2) Lane Closure in California: In the real world, the road is often closed due to road construction. Fig. 3 shows the traffic speeds during two days, which contains traffic dynamics change after lane closure. As shown in Fig. 3, the average speed decreases sharply after starting construction. However, under normal circumstances, the average speed should reach a peak due to the low traffic volume in the middle of the night. Furthermore, the average speeds have a significant increase compared to the normal sequence after finishing construction. Namely, such traffic problem usually impacts the speed first and then causes a significant difference from normal.

In summary, both urban events and traffic incidents have direct influences on traffic data. However, the impacts of

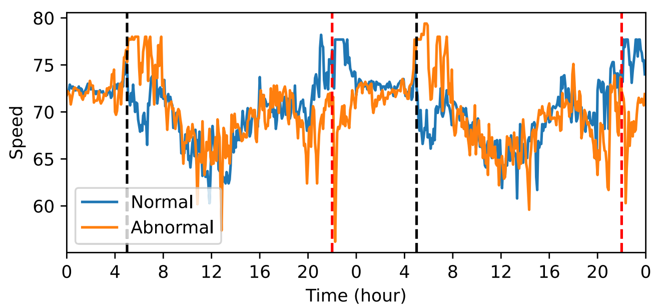


Fig. 3. Traffic speed during two days and the same days of next week. The lane was closed every night from 22:00 to 5:00 for construction. The black dotted line indicates the end time of construction, and the red line indicates the start time.

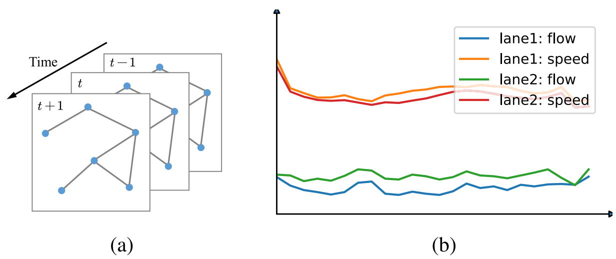


Fig. 4. (a) Graph-structured traffic network. (b) Traffic dynamics of a node on the PeMS dataset. Here, the traffic data are normalized to $[-1, 1]$.

traffic anomalies depend on location, time, and type of traffic anomaly. Therefore, it is necessary to design a flexible and conditional criterion for detecting traffic anomalies. In this work, we propose a spatiotemporal discriminator, which determines whether a data point is abnormal or not depending on its neighbors in spatial and temporal dimensions instead of itself. The discriminator provides a criterion varying with locations and times, which conforms to the characteristics of traffic anomalies in the real world.

IV. Overview

In this section, we introduce basic definitions and an overview of our framework.

A. Preliminaries

1) Traffic Network: In this study, we define a traffic network as a weighted directed graph $G = (V, E, \boldsymbol{W})$ as shown in Fig. 4(a), where V is a finite set of N nodes; E is a set of edges, indicating the connectivity between the nodes; and $\boldsymbol{W} \in \mathbb{R}^{N\times N}$ denotes the weighted adjacency matrix of graph G, representing the spatial correlations between connected nodes. We use a thresholded Gaussian kernel [41] to build the adjacency matrix. The edge weight between nodes υ_i and υ_j is defined as

$$\boldsymbol{W}_{ij} = \begin{cases} \exp\left(-\frac{\mathrm{dist}(\upsilon_i, \upsilon_j)^2}{\sigma^2}\right), & i \neq j \text{ and } e_{i,j} = 1 \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where $\mathrm{dist}(\upsilon_i, \upsilon_j)$ denotes the distance from node υ_i to node υ_j, σ is the standard deviation of distances, e_{ij} represents

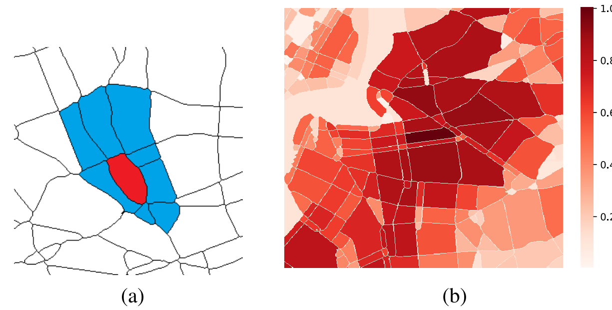


Fig. 5. (a) Subgraph of the central region. (b) Heat map of correlations between the central dark region and other regions.

the edge between nodes υ_i and υ_j, and $e_{i,j} = 1$ if υ_i and υ_j are adjacent and otherwise 0.

2) Traffic Dynamic: Denote the set of time slices as $\mathcal{T}$ and $|\mathcal{T}| = T$. Thus, each data point can be denoted as a triple tuple $s = \langle \upsilon, t, \boldsymbol{x}_{\upsilon,t} \rangle$, where $\upsilon \in V$, $t \in \mathcal{T}$, and $\boldsymbol{x}_{\upsilon,t} \in \mathbb{R}^F$ represents the traffic dynamic of node υ at time t. In the NYC dataset, the traffic dynamics represent the inflows and the outflows of taxis and bikes. In the PeMS dataset, the traffic dynamics represent the total volume and the average speed of each lane. For example, Fig. 4(b) shows a traffic dynamics sequence of a node on the PeMS dataset.

3) Subgraph: There exist high correlations between neighboring locations in the spatial dimension. Fig. 5(b) shows the heatmap of the correlations in the NYC dataset. The central dark region is the object region; the color depths represent the correlations between the object region and other regions. We can find that traffic dynamics of nearby areas are more relevant than ones of distant nodes. Therefore, we model the traffic dynamics of a data point s by considering its neighbors. As shown in Fig. 5(a), the node in red is object node υ, and nodes in blue are its neighboring nodes. We denote the subgraph of node υ as $G_\upsilon = (V_\upsilon, E_\upsilon, \boldsymbol{W}_\upsilon)$, where V_υ consists of υ and its $n-1$ nearest nodes. In this article, we set n to 9 for both two datasets.

4) Traffic Anomaly Detection: Given traffic network G and historical data $\mathcal{S}$, where $\mathcal{S} \in \mathbb{R}^{T\times N\times F}$ denotes the traffic dynamics of all nodes over T time slices, we aim at identifying which nodes are abnormal at time $T+1$.

B. Framework

Fig. 6 shows our proposed framework for STGAN. Our framework consists of two components, i.e., the spatiotemporal generator and the discriminator. As shown in Fig. 6, the sequence, including the generated traffic dynamics, is considered as fake ($\hat{\boldsymbol{S}}_t$), while a real sequence ($\boldsymbol{S}_t$) contains only real traffic dynamics. Through adversarial learning, the spatiotemporal discriminator learns to classify whether the input sequence is real or not. The generator learns to predict traffic dynamics, which can fool the discriminator. We will introduce the framework in detail in the following.

V. Spatiotemporal Generator

In this section, we introduce the architecture of the generator in detail. It consists of three independent modules,

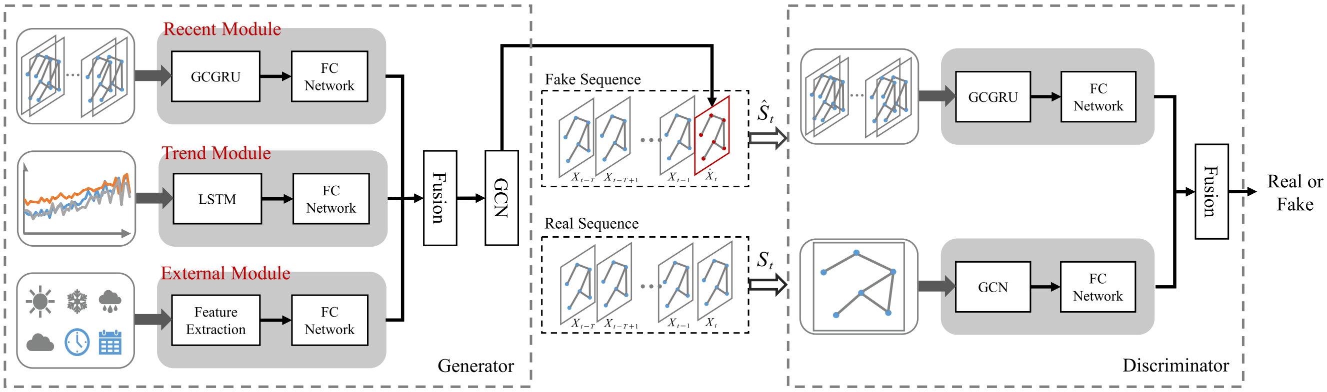
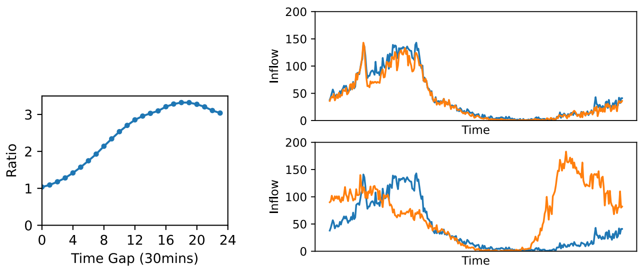


Fig. 6. Framework of STGAN. Left: spatiotemporal generator. Right: spatiotemporal discriminator.



(a) (b)

Fig. 7. (a) Temporal dependencies. (b) Spatial dependencies between two nodes. The top part is a pair adjacent nodes and the bottom part is a pair of distant nodes.

including the recent, trend, and external modules, to model the spatiotemporal, trend, and external features. In the end, the outputs of three modules are further merged based on a graph convolution network (GCN) layer to obtain the final prediction result.

A. Recent Module

The recent module is proposed to capture the high correlations between traffic dynamics of neighboring data points in the spatial and temporal dimensions concurrently. It consists of a graph convolutional gated recurrent unit (GCGRU) [42] layer and a fully connected layer.

We visualize the real traffic dynamics to demonstrate the high correlations between neighbors in the spatial and temporal dimensions in Fig. 7. Fig. 7(a) shows the average ratio curves using the NYC dataset, where the x-axis indicates the time gap and the y-axis indicates the average ratio between two taxi inflows having the corresponding time gap. It can be seen that the traffic dynamics of recent time intervals are more relevant than ones of distant time intervals. Fig. 7(b) compares the time series of one node with its adjacent node and distant node using the PeMS dataset. We can find that the adjacent nodes have similar traffic dynamics, and the distant nodes have two distinct traffic dynamics.

1) Recent Segment: In this article, we capture spatiotemporal dependencies in traffic data by introducing a recent segment. We use T_r to denote the length of the input of the recent module. Thus, assuming that the current time is t, the recent segment of the data point $s = \langle v, t, \boldsymbol{x}_{v,t}\rangle$ can be formulated as follows:

$$\boldsymbol{\mathcal{X}}_r = \left(\boldsymbol{X}_{v,t-T_r}, \boldsymbol{X}_{v,t-T_r+1}, \ldots, \boldsymbol{X}_{v,t-1}\right) \in \mathbb{R}^{T_r \times n \times F} \tag{2}$$

where $\boldsymbol{X}_{v,t} = \{\boldsymbol{x}_{v_i,t} | v_i \in G_v\}$ denotes the traffic dynamics of the subgraph of node v during time interval t.

2) Graph Convolutional Gated Recurrent Unit: In this article, we denote a traffic network as a weighted directed graph. The standard convolution for the image or regular grids cannot be directly applicable to graph structure [43]. Therefore, we leverage spectral graph theory, which generalizes the traditional convolution operation to the graph structure data. Given a graph $G = (V, E, \boldsymbol{W})$, let $\boldsymbol{L} = \boldsymbol{I} - \boldsymbol{D}^{-(1/2)}\boldsymbol{W}\boldsymbol{D}^{-(1/2)} \in \mathbb{R}^{N\times N}$ denote the graph Laplacian matrix, where $\boldsymbol{I}$ is an identity matrix and $\boldsymbol{D}$ is the degree matrix. Then, a graph convolution operation can be defined as

$$\boldsymbol{\Theta}_g *_G \boldsymbol{X} = \mathrm{relu}\left(\boldsymbol{L}\boldsymbol{X}\boldsymbol{\Theta}_g + \boldsymbol{b}_g\right) \tag{3}$$

where $*_G$ denotes a graph convolution operation and $\boldsymbol{\Theta}_g$ and $\boldsymbol{b}_g$ are learnable parameters.

In order to model spatial and temporal dependencies concurrently, we devise the recent module, which mainly consists of the GCGRU [42]. The GCGRU replaces the matrix multiplications in gated recurrent unit (GRU) [44] with the graph convolution, and then, the GCGRU can be defined as follows:

$$\begin{aligned}
\boldsymbol{r}^{(t)} &= \sigma\left(\boldsymbol{\Theta}_r *_G \left[\boldsymbol{X}^{(t)}, \boldsymbol{H}^{t-1}\right] + \boldsymbol{b}_r\right)\\
\boldsymbol{u}^{(t)} &= \sigma\left(\boldsymbol{\Theta}_u *_G \left[\boldsymbol{X}^{(t)}, \boldsymbol{H}^{t-1}\right] + \boldsymbol{b}_u\right)\\
\boldsymbol{C}^{(t)} &= \tanh\left(\boldsymbol{\Theta}_C *_G \left[\boldsymbol{X}^{(t)}, \left(\boldsymbol{r}^{(t)}\ \boldsymbol{H}^{t-1}\right)\right] + \boldsymbol{b}_c\right)\\
\boldsymbol{H}^{(t)} &= \boldsymbol{u}^{(t)}\ \boldsymbol{H}^{(t-1)} + \left(1 - \boldsymbol{u}^{(t)}\right)\ \boldsymbol{C}^{(t)}
\end{aligned} \tag{4}$$

where $\boldsymbol{X}^{(t)}$ and $\boldsymbol{H}^{(t)}$ denote the input and output at time t, respectively, $\boldsymbol{r}^{(t)}$ and $\boldsymbol{u}^{(t)}$ are reset gate and update gate at time t, respectively, $*_G$ denotes the graph convolution defined in (3), and $\boldsymbol{\Theta}_r, \boldsymbol{\Theta}_u, \boldsymbol{\Theta}_C, \boldsymbol{b}_r, \boldsymbol{b}_u$, and $\boldsymbol{b}_c$ are learnable parameters [45].

B. Trend Module

As visualized in Fig. 7, the spatiotemporal features of data points are more relevant to the short-term traffic dynamics.

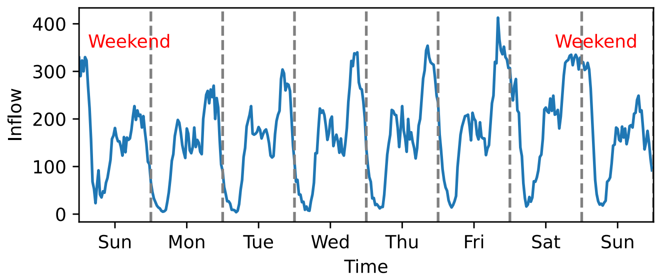


Fig. 8. Taxi inflows during weekend and weekday.

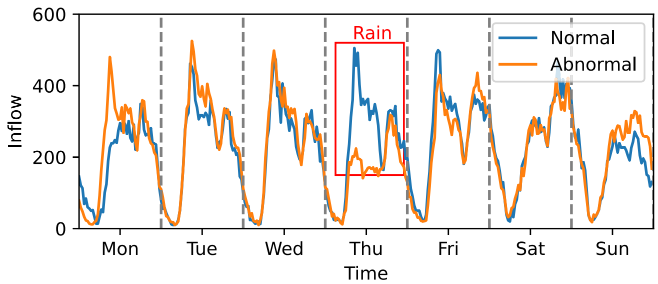


Fig. 9. Taxi inflows during normal days and rainy day in the same region.

Therefore, the generator learns the spatiotemporal features only depending on the recent segment defined in (2). Then, the trend module aims at learning the long-term temporal features. We use T_h to denote the input length of the trend segment. Thus, assuming that the current time is t, the trend segment can be formulated as follows:

$$\boldsymbol{\mathcal{X}}_h = \left(\boldsymbol{x}_{v,t-T_h}, \boldsymbol{x}_{v,t-T_h+1}, \ldots, \boldsymbol{x}_{v,t-1}\right) \in \mathbb{R}^{T_h \times F}. \quad (5)$$

The trend module contains an LSTM layer and a fully connected layer. By the LSTM layer, the trend of traffic data of each node can be learned.

C. External Module

The external module is proposed to learn how external feature affects traffic data. As shown in Fig. 6, the external module consists of a fully connected layer, which takes the external features as input.

Traffic flows can be affected by many complex external factors, such as time and weather. First, we consider the effect of time factors. Fig. 8 shows taxi inflow during eight days in NYC. It can be seen that the traffic flows show similar trends every day. For instance, the traffic flow decreases sharply at midnight and increases in the morning. Moreover, the traffic flows on weekdays show different trends from weekends. Second, we consider the effect of weather factors. Fig. 9 presents taxi inflow during two weeks, and the weather of one day is rain. We find that bad weather causes a sharp decrease compared to the same day of next week. In this article, we represent the external feature as follows:

$$E = \left[\boldsymbol{O}_{\text{weekday}};\ \boldsymbol{O}_{\text{hour}};\ \boldsymbol{O}_{\text{weather}}\right] \quad (6)$$

where $\boldsymbol{O}_{\text{weekday}}$ is a one-hot vector of length 7 representing the day of week, $\boldsymbol{O}_{\text{hour}}$ is a one-hot vector of length 24 that shows which hour it is in a day, and $\boldsymbol{O}_{\text{weather}}$ is also a one-hot vector denoting the weather. Some incidents on the freeway are caused by bad weather. Therefore, we do not consider weather features on the PeMS dataset.

D. Fusion

For each data point $s = \langle v, t, \boldsymbol{x}_{v,t}\rangle$, the goal of the generator is to predict the traffic dynamics of subgraph G_v, that is, X_v. Its three modules aim at learning the short-term spatiotemporal feature, long-term temporal feature, and the external feature of s, respectively. Due to the high correlations between adjacent nodes, the spatial feature of node v can be used to reconstruct the traffic dynamics of subgraph G_v via a GCN layer. The final prediction of s can be defined as follows:

$$\hat{\boldsymbol{X}}_{v,t} = \tanh(\boldsymbol{\Theta}_F *_G [\boldsymbol{X}_R, \boldsymbol{X}_T, \boldsymbol{X}_E,]) \quad (7)$$

where $*_G$ denotes the graph convolution defined in (3) and $\boldsymbol{X}_R$, $\boldsymbol{X}_T$, and $\boldsymbol{X}_E$ denote the outputs of the recent module, the trend module, and the external module, respectively.

E. Loss Function

Let $G_{\boldsymbol{\theta}}$ and $D_{\boldsymbol{\phi}}$ denote the generator function with parameter $\boldsymbol{\theta}$ and the discriminator function with parameter $\boldsymbol{\phi}$, respectively. To generate realistic instances, we design a loss function composed of prediction error and discriminator loss, which can be formulated as follows:

$$\mathcal{L}_G(\boldsymbol{\theta}) = \sum_s - \log\left(1 - D_{\boldsymbol{\phi}}\left(\hat{\boldsymbol{S}}_{v,t}\right)\right) + \lambda_G \|G_{\boldsymbol{\theta}}(v,t) - \boldsymbol{X}_{v,t}\|_2 \quad (8)$$

where $\hat{\boldsymbol{S}}_{v,t} = \{\boldsymbol{X}_{v,t-T_r}, \boldsymbol{X}_{v,t-T_r+1}, \ldots, \hat{\boldsymbol{X}}_{v,t-1}\}$ denotes the fake sequence and λ_G is a hyperparameter to balance the prediction error and the discriminator loss. By minimizing the two parts, the generator will generate realistic instances, which can fool the discriminator.

VI. Spatiotemporal Discriminator

In this section, we first introduce the structure of the discriminator in detail and then propose a flexible anomaly score to combine the detection performance of the generator and the discriminator. The spatiotemporal discriminator consists of a GCN layer and a GCGRU layer to model the spatial features at the current time and the spatiotemporal features, respectively. In the end, the outputs of two modules are merged based on a fully connected layer to obtain the classification result.

A. Spatiotemporal Discriminator

As shown in Fig. 6, the GCN layer defined in (3) is used to capture the spatial features at the current time. The structure of the GCGRU layer defined in (4) is used to learn the spatiotemporal features from the recent segment, which is similar to the generator component. Finally, the fully connected layer uses sigmoid as the activation function and outputs anomaly scores in [0, 1] for each input sequence. Note that the scores of anomalies are 1 and vice versa are 0.

To distinguish the real sequence from the fake sequence, we design an adversarial loss. By minimizing the loss, the discriminator can identify whether the input sequence is real or not. The loss function can be written as

$$\mathcal{L}_D(\boldsymbol{\phi}) = \sum_s -\log\big(D_{\boldsymbol{\phi}}(\hat{\boldsymbol{S}}_{v,t})\big) - \log\big(1 - D_{\boldsymbol{\phi}}(\boldsymbol{S}_{v,t})\big) \quad (9)$$

where $\boldsymbol{S}_{v,t} = \{\boldsymbol{X}_{v,t-T_r}, \boldsymbol{X}_{v,t-T_r+1}, \ldots, \boldsymbol{X}_{v,t-1}, \boldsymbol{X}_{v,t}\}$ and $\hat{\boldsymbol{S}}_{v,t} = \{\boldsymbol{X}_{v,t-T_r}, \boldsymbol{X}_{v,t-T_r+1}, \ldots, \boldsymbol{X}_{v,t-1}, \hat{\boldsymbol{X}}_{v,t}\}$ are the real and fake sequences, respectively.

During the adversarial training, the discriminator can promote the learning of the generator and train with the generated negative data. After the end, the discriminator can be used as a detector.

B. Anomaly Score

Next, we need to define an anomaly score to detect anomalies. As we said before, both the generator and the discriminator can be used as detectors independently. For the generator, the anomaly score can be designed as a prediction error, which detects the anomalies with sudden changes in traffic data. For the discriminator, the output can be used as the anomaly score directly, which detects the anomalies deviating from spatiotemporal features. Obviously, we can combine the detection abilities of these two components [38].

However, at the end of the training, we find that the generator can generate a realistic instance to fool the discriminator. Thus, the discriminator cannot determine the generated data with high confidence. Meanwhile, the traffic data in real world are affected by various complex factors. Therefore, the real traffic dynamics would present more unpredictable patterns than the corresponding generated data regardless of normal or abnormal. Based on the above findings, we design a flexible anomaly score to reduce the impact of the unpredictability of traffic dynamics. Instead of directly using the discriminator score of real sequence, we compute the difference value between the discriminator scores of real and fake sequences. For a data point $s = \langle v, t, \boldsymbol{x}_{v,t}\rangle$, the anomaly score can be written as follows:

$$\begin{aligned} s_D(v,t) &= D_{\boldsymbol{\phi}}(\boldsymbol{S}_{v,t}) - D_{\boldsymbol{\phi}}(\hat{\boldsymbol{S}}_{v,t}) \\ s_G(v,t) &= \|G_{\boldsymbol{\theta}}(v,t) - \boldsymbol{X}_{v,t}\|_2 \\ \text{score}(v,t) &= s_G(v,t) + \lambda s_D(v,t) \end{aligned} \quad (10)$$

where λ is a hyperparameter to balance the generator detection and the discriminator detection.

VII. Experiments

In this section, we evaluate the proposed spatiotemporal anomaly detection method on two real datasets and compare it with the existing works, including individual methods, temporal methods, and spatiotemporal methods. Besides comparing the detection performances, we also conduct the prediction performance comparison and ablation experiments.

A. Ground Truth Data

1) PeMS Dataset: The Caltrans PeMS provides California Highway Patrol (CHP) incident reports and Lane Closure System (LCS) reports. The CHP records the start time, duration, location, and type of some traffic incidents that occurred in real world.[5] The LCS records the start time, end time, start location, end location, and type of some lane closure incidents. We collect 458 CHP incident reports and 78 LCS reports from April 16, 2014 to May 6, 2014. In the experiments, we extend the end time of each CHP incident by 1 h to include the impact of the traffic accidents. Note that we do not consider the weather feature in the external module, so we do not filter the incidents cause by weather. The statistics of the ground truth are summarized in Table II.

TABLE II
GROUND TRUTH ON THE PEMS DATASET

Properties	CHP Incident Report	LCS Report
Number Of Categories	18	13
Number Of Instances	458	78
Average Duration (mins)	33.9	468.7
Average Distance (post mile)	/	1.34

2) NYC Dataset: The traffic anomalies in cities are generally caused by social events. Therefore, we collect 20 events from November 1, 2014 to November 30, 2014 as ground truth.[6] For example, Christmas Tree Lighting, 2017/12/1 18:00-20:00, and Bryant Park denote the description, time, and location of one recorded event, respectively.

B. Evaluation Metrics

Evaluating anomaly detection in real-world settings is an open challenge since it is difficult to obtain a complete set of ground truth. Following the same procedure in previous work [32], [40], [46], we employ recall to measure the detection performances. Since the reported incidents are not a complete set of ground truth in real world, we do not use precision as the evaluation metric. We represent the set of real-world anomalies in ground truth as Θ. We mark the data points given the $K\%$ highest anomaly scores as anomalies, represented by Θ_K. For each detected anomaly, if a real-world anomaly spatially and temporally overlaps with the detected anomaly, we consider the detection as a hit. Thus, the recall is calculated as follows:

$$\text{Recall@K\%} = \frac{|\Theta \cap \Theta_K|}{|\Theta|}. \quad (11)$$

C. Baselines and Settings

To prove the effectiveness of STGAN, we evaluate three groups of the anomaly detection method, namely, individual methods, temporal methods, and spatiotemporal methods. A detailed description of these baselines is illustrated as follows.

1) Individual Methods: Individual methods (isolation forest (IF) [47], elliptic envelope (EE) [48], and LOF [49]) and random forest (RF) [50] detect anomalies from traffic dynamics, ignoring the dependencies between data points. In the

[5]https://pems.dot.ca.gov/Transit_PeMS_User Manual_v1.0.pdf
[6]https://www.nycinsiderguide.com/

experiments, we train models for each node independently and then normalize and merge the anomaly scores of all nodes to decide the final anomalies.

1) IF [47] constructs binary trees based on the attributes of data points to isolate abnormal points. The data points with the smaller path length in the tree are considered as anomalies.
2) EE [48] fits the data points with an EE and defines the anomaly score as Mahalanobis distance.
3) LOF [49] considers the data points whose local densities are significantly lower than neighbors as anomalies.
4) RF [50] is an ensemble learning method for classification. In the experiments, due to the lack of labeled data during training, we generate the fake labeled data to transform the unsupervised anomaly detection into a supervised problem.

2) Temporal Methods: Temporal methods [vector autoregressive model (VAR) [51] and preprocessing VAR (pVAR)] predict traffic dynamics only considering the temporal features and then regard the prediction error as the anomaly score.

1) VAR [52] is one of the most popular linear models for time-series forecasting.
2) pVAR applies the VAR model to preprocessed data, which is decomposed from the traffic data of each node.

3) Spatiotemporal Methods: Spatiotemporal anomaly detection methods (ST_decompn [32], CIAS&AIAS [3], and TBAD [28]), include tensor factorization-based, deep learning-based, and statistical models.

1) ST_decompn [32] is an urban anomaly detection model based on decomposition. More specifically, this method learns the spatial and temporal features separately in advance, then models the normal traffic patterns using a deep neural network. The LOF algorithm is applied to the abnormal components to get final anomaly scores.
2) CIAS&AIAS [3] estimates the anomaly score for each region based on its historically similar regions. Then, this work leverages OC-SVM to compute the final anomaly score. This method outputs a set of abnormal data points instead of anomaly scores. Therefore, in the experiment, we compute the recall and the corresponding anomaly ratio.
3) ASTGCN [53] is a state-of-the-art traffic flow prediction method. In the experiments, we use ASTGCN to predict the traffic dynamics instead of traffic flow and define the anomaly score as the prediction error.
4) TBAD [28] is a tensor factorization-based method, which builds up a set of tensors to represent daily traffic dynamics and then applies nonnegative CP decomposition on the tensors. During testing, the regions whose current spatial features deviate from history are regarded as anomalies. In the experiments, we combine the spatial and temporal feature vectors to represent a data point and then leverage the LOF algorithm to detect anomalies.

D. Settings

In the experiments, we use the min-max normalization method to scale the traffic data into the range $[-1, 1]$. The

TABLE III
RECALL@K% ON THE PEMS DATASET

Method	Recall@K% 1%	2%	3%	4%	10%
IF	0.06	0.08	0.11	0.15	0.32
EE	0.05	0.07	0.09	0.09	0.20
LOF	0.15	0.31	0.41	0.49	0.68
RF	0.15	0.31	0.39	0.45	0.70
VAR	0.21	0.30	0.36	0.39	0.62
pVAR	0.21	0.30	0.36	0.41	0.64
ASTGCN	0.16	0.24	0.31	0.37	0.54
TBAD	0.18	0.31	0.40	0.50	0.70
ST-Decompn	0.22	0.36	0.45	0.53	0.74
STGAN	**0.24**	**0.40**	**0.51**	**0.58**	**0.82**

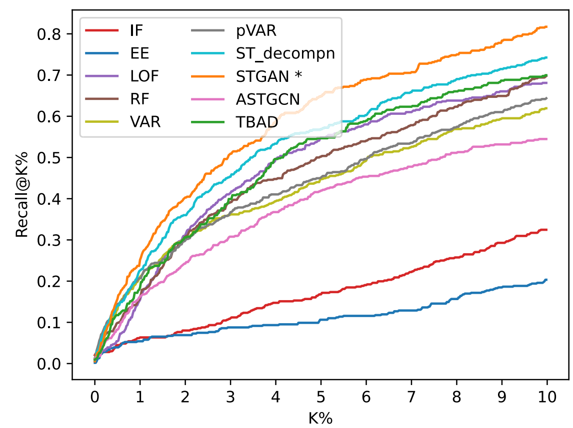


Fig. 10. Recall curve @K% on the PeMS dataset.

hyperparameter λ_G in (8) is set as 500. For the anomaly score defined in (10), we normalize $s_D(v, t)$ and $s_G(v, t)$ and then set $\lambda = 1$. The time length of the recent segment is 1 h on the PeMS dataset and 2 h on the NYC dataset. We set a batch size as 256 and use Adam [54] with the learning rate of 0.001 to train the model. We use the LSTM network with depth 2 and 64 hidden units for the generator. The GCGRU networks for the generator and the discriminator use two layers and 32 hidden units. For all the other parts of the network, the number of hidden units is set to 64. The source code is available at https://github.com/dleyan/STGAN.

E. Results

1) PeMS Dataset: For the PeMS dataset, we use the data of the last 20 days for testing and the others for training. The baseline CIAS&AIAS detects 1.5% anomalies and achieves Recall = 0.10. Table III and Fig. 10, respectively, show the recalls and recall curves of different methods. It can be seen that our method STGAN achieves the best performances with different values of K over recall. In addition, we have some other findings.

1) The spatiotemporal methods except for ASTGCN show better performances than individual methods and temporal methods. These results prove that it is important to model the spatiotemporal features of traffic data.

TABLE IV
HIT EVENTS@K% ON THE NYC DATASET

Method	Hit events@K% 0.2%	0.4%	0.6%	0.8%	1%
IF	3	5	11	13	13
EE	5	9	11	11	11
LOF	2	8	15	15	15
RF	11	13	14	15	15
VAR	9	12	12	13	13
pVAR	9	13	15	15	15
ASTGCN	8	11	11	14	15
TBAD	**12**	12	13	14	15
ST_decompn	2	8	15	15	17
STGAN	10	**16**	**17**	**17**	**19**

2) The experimental results show that our method outperforms the recently proposed ST_decompn. The reason is that ST_decompn extracts spatial and temporal features from historical data separately, which affects the ability to model traffic dynamics.
3) Our method significantly outperforms ASTGCN, demonstrating that detecting traffic anomalies based on spatiotemporal features is necessary.
4) The temporal method pVAR improves the performance of VAR, which validates that traffic data present periodical patterns, and the external module based on time features is reasonable.

2) NYC Dataset: For the NYC dataset, we use the data of last month for testing and the others for training. The baseline CIAS&AIAS detects 1.0% anomalies and achieves hit events = 10. Table IV shows the number of hit events detected by other methods. First, we find that STGAN achieves the highest recall in most cases except for $K = 0.2$. Moreover, TBAD only achieves the best performance when $K = 0.2$, but the recall improves slowly with the increase of detected anomalies. Second, the temporal methods show a similar performance compared to spatiotemporal methods. One possible reason is that the social event often causes a sharp increase in urban flow, as shown in Fig. 2. Thus, the temporal methods based on prediction accuracy can effectively detect abnormal events. Third, ST_decompn achieves the second-best overall results but presents extremely poor performance when $K = 0.2$. The results illustrate that the anomalies detected by ST_decompn with high confidences are not accurate.

3) Prediction Performance: A common key idea in existing works is detecting anomalies relying on prediction tasks. Therefore, we conduct the prediction accuracy comparison for the spatiotemporal methods. Note that we only evaluate the prediction errors of central nodes for the proposed STGAN. Moreover, limited by the size of the service area of taxis and bicycles, the NYC dataset is sparse. The missing values in the PeMS dataset are filled by zero, also resulting in sparseness. Therefore, the differences between the prediction errors of different generators are not significant enough. Table VII shows the prediction errors on two real-world datasets. As shown in Table VII, our proposed STGAN achieves the best prediction performance of these three spatiotemporal methods. Compared to ASTGCN, our proposed STGAN learns the spatiotemporal features of data points from their neighbors in spatial and temporal dimensions, and these neighbors are demonstrated to be highly correlated with data points, as shown in Fig. 7. Moreover, the other two methods outperform ST_decompn significantly. The reason is that ST_decompn constructs fixed spatial features and temporal features in advance. Then, they merge these features via a neural network. Thus, it cannot learn the dynamic spatiotemporal features well.

TABLE V
GENERATOR PREDICTION ERROR

Dataset / Method	NYC dataset RMSE	NYC dataset MAE	PeMS dataset RMSE	PeMS dataset MAE
No Recent Module	19.30	4.14	17.11	9.64
No External Module	5.68	1.47	5.11	2.21
No Trend Module	5.70	1.36	5.15	2.25
Proposed Generator	**5.55**	**1.34**	**5.07**	**2.20**

TABLE VI
ANOMALY SCORE PERFORMANCE

Anomaly Score	Recall@K% 1%	2%	3%	4%	10%
$D(\boldsymbol{S}_{v,t})$	0.2071	0.3022	0.3825	0.4403	0.6866
$s_D(v,t)$	**0.2425**	0.3843	0.4888	0.5802	**0.8172**
$s_G(v,t)$	0.2071	0.3843	0.4757	0.5765	0.7929
$\text{score}(v,t)$	0.2369	**0.4011**	**0.5075**	**0.5840**	**0.8172**

TABLE VII
SPATIOTEMPORAL METHOD PREDICTION ERROR

Dataset / Method	NYC dataset RMSE	NYC dataset MAE	PeMS dataset RMSE	PeMS dataset MAE
ASTGCN	6.70	2.06	7.54	3.62
ST_decompn	15.03	3.76	25.21	12.40
STGAN	**5.55**	**1.34**	**5.07**	**2.20**

F. Anomaly Score Analysis

To demonstrate the effectiveness of the proposed flexible anomaly score, we compare the detection performances with others in Table VI. Obviously, the proposed anomaly score achieves the overall best performances. In addition, we have some other findings. First, comparing the performances of $s_D(v,t)$ with $D(\boldsymbol{S}_{v,t})$, we can see the significant improvements, which proves that the difference can fix the detection of a single discriminator score. Second, $s_G(v,t)$ performs weaker than the proposed $\text{score}(v,t)$ but better than the state-of-the-art traffic flow prediction method ASTGCN. The possible reason is that the proposed generator not only aims at reducing prediction error but also models the normal traffic patterns.

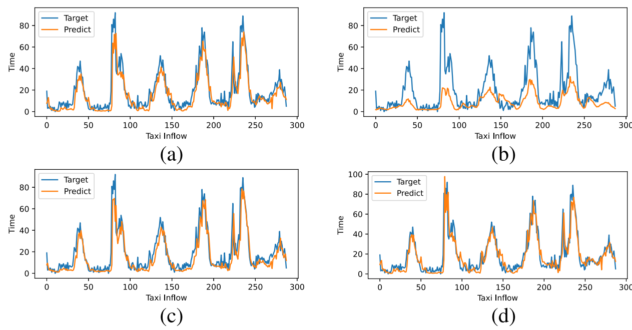


Fig. 11. Prediction visualization of different generators. (a) Generator without trend module. (b) Generator without recent module. (c) Generator without external module. (d) Proposed generator.

G. Module Analysis

The generator consists of three independent modules. To demonstrate the necessity of each module, we conduct prediction comparisons of the generator without different modules on two real datasets. Note that the goal of the generator is to predict traffic dynamics of the subgraph, but we evaluate the prediction performances only using central nodes. It can be seen from Table V that the complete generator achieves the best prediction performances on two datasets. We further visualize the prediction of the different generators in Fig. 11. We can first observe that the generator without the recent module is unable to predict traffic dynamics well. The reason is that the generator aims at predicting traffic dynamics of nodes in the subgraph, and only the recent module can model the spatial features. Second, the generator can only learn the long-term from external information. Therefore, it presents prediction delay and high prediction error during sudden changes in Fig. 11(a). Similarly, the generator without the external module also shows delays in prediction. However, it predicts the sudden changes more accurately. Compared to the three ablated generators, the proposed generator better fits the real traffic dynamics and achieves the best prediction accuracy.

VIII. Conclusion

With the rapid growth of sensors and mobile devices, there are a variety of traffic data. The massive traffic data can help detect abnormal traffic events. In this article, we proposed a spatiotemporal graph convolutional adversarial framework called STGAN to detect traffic anomalies. The proposed generator consists of three modules to model the normal traffic dynamics patterns. The spatiotemporal generator and discriminator have a similar module using GCGRU. The GCGRU captures the high correlations between neighboring points in spatial and temporal dimensions and helps the generator and the discriminator learn the spatiotemporal features of traffic data and traffic anomalies, respectively. Then, we devised a flexible anomaly score varying with locations and times for traffic anomaly detection. Evaluations on two real-world datasets from NYC and California demonstrate that our framework effectively detects traffic anomalies and outperforms baselines.

In the future, we plan to extend the proposed discriminator to classify the categories of traffic anomalies.

References

[1] Z. Yu, C. Licia, W. Ouri, and Y. Hai, "Urban computing: Concepts, methodologies, and applications," *ACM Trans. Intell. Syst. Technol.*, vol. 5, no. 3, pp. 1–55, Sep. 2014.
[2] M. Zhang, T. Li, Y. Yu, Y. Li, P. Hui, and Y. Zheng, "Urban anomaly analytics: Description, detection and prediction," *IEEE Trans. Big Data*, early access, Apr. 28, 2020, doi: 10.1109/TBDATA.2020.2991008.
[3] H. Zhang, Y. Zheng, and Y. Yu, "Detecting urban anomalies using multiple spatio-temporal data sources," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 1, pp. 1–18, Mar. 2018.
[4] Z. Fan, A. Arai, X. Song, A. Witayangkurn, H. Kanasugi, and R. Shibasaki, "A collaborative filtering approach to citywide human mobility completion from sparse call records," in *Proc. IJCAI*, 2016, pp. 2500–2506.
[5] C. Chen, D. Zhang, Z.-H. Zhou, N. Li, T. Atmaca, and S. Li, "B-planner: Night bus route planning using large-scale taxi GPS traces," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*. Mar. 2013, pp. 225–233.
[6] *Melbourne 2030: A Planning Update Melbourne@ 5 Million*, Dept. Planning Community Develop., Melbourne, VIC, Australia, 2008.
[7] V. Garcia-Font, C. Garrigues, and H. Rifà-Pous, "A comparative study of anomaly detection techniques for smart city wireless sensor networks," *Sensors*, vol. 16, no. 6, p. 868, Jun. 2016.
[8] D. Zhang, B. Guo, and Z. Yu, "The emergence of social and community intelligence," *Computer*, vol. 44, no. 7, pp. 21–28, Jul. 2014.
[9] R. Ji, Y. Gao, W. Liu, X. Xie, Q. Tian, and X. Li, "When location meets social multimedia: A survey on vision-based recognition and mining for geo-social multimedia analytics," *ACM Trans. Intell. Syst. Technol.*, vol. 6, no. 1, pp. 1–18, Mar. 2015.
[10] B. Ubaldi, "Open government data: Towards empirical analysis of open government data initiatives," OECD, Paris, France, Tech. Rep. 22, 2013.
[11] M. C. González, C. A. Hidalgo, and A.-L. Barabási, "Understanding individual human mobility patterns," *Nature*, vol. 453, no. 7196, pp. 779–782, 2008.
[12] C. Zhuang, N. J. Yuan, R. Song, X. Xie, and Q. Ma, "Understanding people lifestyles: Construction of urban movement knowledge graph from GPS trajectory," in *Proc. IJCAI*, 2017, pp. 3616–3623.
[13] W. Zhang, G. Qi, G. Pan, H. Lu, S. Li, and Z. Wu, "City-scale social event detection and evaluation with taxi traces," *ACM Trans. Intell. Syst. Technol.*, vol. 6, no. 3, pp. 1–20, May 2015.
[14] L. Chen, D. Yang, J. Jakubowicz, G. Pan, D. Zhang, and S. Li, "Sensing the pulse of urban activity centers leveraging bike sharing open data," in *Proc. IEEE 12th Int. Conf. Ubiquitous Intell. Comput., IEEE 12th Int. Conf. Autonomic Trusted Comput., IEEE 15th Int. Conf. Scalable Comput. Commun. Associated Workshops (UIC-ATC-ScalCom)*, Aug. 2015, pp. 135–142.
[15] M.-F. Chiang, E.-P. Lim, W.-C. Lee, and A. T. Kwee, "BTCI: A new framework for identifying congestion cascades using bus trajectory data," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 1133–1142.
[16] J. Zhang, "Smarter outlier detection and deeper understanding of large-scale taxi trip records: A case study of NYC," in *Proc. ACM SIGKDD Int. Workshop Urban Comput.*, 2012, pp. 157–162.
[17] Y. Ge, H. Xiong, Z.-H. Zhou, H. Ozdemir, J. Yu, and K. C. Lee, "TOP-EYE: Top-*k* evolving trajectory outlier detection," in *Proc. 19th ACM Int. Conf. Inf. Knowl. Manage.*, 2010, pp. 1733–1736.
[18] X. Li, Z. Li, J. Han, and J.-G. Lee, "Temporal outlier detection in vehicle traffic data," in *Proc. IEEE 25th Int. Conf. Data Eng.*, Mar. 2009, pp. 1319–1322.
[19] Y. Wang, J. Xu, M. Xu, N. Zheng, J. Jiang, and K. Kong, "A feature-based method for traffic anomaly detection," in *Proc. 2nd ACM SIGSPATIAL Workshop Smart Cities Urban Anal.*, 2016, pp. 1–8.
[20] S. Yang, K. Kalpakis, and A. Biem, "Detecting road traffic events by coupling multiple timeseries with a nonparametric Bayesian method," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 1936–1946, Mar. 2014.
[21] I. Ceapa, C. Smith, and L. Capra, "Avoiding the crowds: Understanding tube station congestion patterns from trip data," in *Proc. ACM SIGKDD Int. Workshop Urban Comput.*, 2012, pp. 134–141.
[22] R. Mehran, A. Oyama, and M. Shah, "Abnormal crowd behavior detection using social force model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 935–942.

[23] L. X. Pang, S. Chawla, W. Liu, and Y. Zheng, "On mining anomalous patterns in road traffic streams," in *Proc. Int. Conf. Adv. Data Mining Appl.* Berlin, Germany: Springer, 2011, pp. 237–251.

[24] S. Yang and W. Liu, "Anomaly detection on collective moving patterns: A hidden Markov model based solution," in *Proc. Int. Conf. Internet Things 4th Int. Conf. Cyber, Phys. Social Comput.*, Oct. 2011, pp. 291–296.

[25] A. Witayangkurn, T. Horanont, Y. Sekimoto, and R. Shibasaki, "Anomalous event detection on large-scale GPS data from mobile phones using hidden Markov model and cloud platform," in *Proc. ACM Conf. Pervasive Ubiquitous Comput. Adjunct Publication*, Sep. 2013, pp. 1219–1228.

[26] L. Sun and K. W. Axhausen, "Understanding urban mobility patterns with a probabilistic tensor factorization framework," *Transp. Res. B, Methodol.*, vol. 91, pp. 511–524, Sep. 2016.

[27] Z. Fan, X. Song, and R. Shibasaki, "CitySpectrum: A non-negative tensor factorization approach," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput.*, Sep. 2014, pp. 213–223.

[28] C. Lin, Q. Zhu, S. Guo, Z. Jin, Y.-R. Lin, and N. Cao, "Anomaly detection in spatiotemporal data via regularized non-negative tensor analysis," *Data Mining Knowl. Discovery*, vol. 32, no. 4, pp. 1056–1073, Jul. 2018.

[29] L. Chen, J. Jakubowicz, D. Yang, D. Zhang, and G. Pan, "Fine-grained urban event detection and characterization based on tensor cofactorization," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 3, pp. 380–391, Jun. 2017.

[30] J. Wang, J. Wu, Z. Wang, F. Gao, and Z. Xiong, "Understanding urban dynamics via context-aware tensor factorization with neighboring regularization," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 11, pp. 2269–2283, Nov. 2020.

[31] H. D. Trinh, L. Giupponi, and P. Dini, "Urban anomaly detection by processing mobile traffic traces with LSTM neural networks," in *Proc. 16th Annu. IEEE Int. Conf. Sens., Commun., Netw. (SECON)*, Jun. 2019, pp. 1–8.

[32] M. Zhang, T. Li, H. Shi, Y. Li, and P. Hui, "A decomposition approach for urban anomaly detection across spatiotemporal data," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 6043–6049.

[33] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[34] G. Pang, C. Shen, L. Cao, and A. van den Hengel, "Deep learning for anomaly detection: A review," 2020, *arXiv:2007.02500*.

[35] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *Proc. Int. Conf. Inf. Process. Med. Imag.* Berlin, Germany: Springer, 2017, pp. 146–157.

[36] S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon, "Ganomaly: Semi-supervised anomaly detection via adversarial training," in *Proc. Asian Conf. Comput. Vis.* Berlin, Germany: Springer, 2018, pp. 622–637.

[37] D. Li, D. Chen, B. Jin, L. Shi, J. Goh, and S.-K. Ng, "Mad-GAN: Multivariate anomaly detection for time series data with generative adversarial networks," in *Proc. Int. Conf. Artif. Neural Netw.* Berlin, Germany: Springer, 2019, pp. 703–716.

[38] S. Lee, H. G. Kim, and Y. M. Ro, "STAN: Spatio-temporal adversarial networks for abnormal event detection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 1323–1327.

[39] M. Ravanbakhsh, M. Nabi, E. Sangineto, L. Marcenaro, C. Regazzoni, and N. Sebe, "Abnormal event detection in videos using generative adversarial nets," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 1577–1581.

[40] Y. Zheng, H. Zhang, and Y. Yu, "Detecting collective anomalies from multiple spatio-temporal datasets across different domains," in *Proc. 23rd SIGSPATIAL Int. Conf. Adv. Geograph. Inf. Syst.*, Nov. 2015, pp. 1–10.

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

[42] Y. Seo, M. Defferrard, P. Vandergheynst, and X. Bresson, "Structured sequence modeling with graph convolutional recurrent networks," in *Proc. Int. Conf. Neural Inf. Process.* Berlin, Germany: Springer, 2018, pp. 362–373.

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

[44] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014, *arXiv:1412.3555*.

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

[46] B. Pan, Y. Zheng, D. Wilkie, and C. Shahabi, "Crowd sensing of traffic anomalies based on human mobility and social media," in *Proc. 21st ACM SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, Nov. 2013, pp. 344–353.

[47] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. 8th IEEE Int. Conf. Data Mining*, Dec. 2008, pp. 413–422.

[48] P. J. Rousseeuw and K. Van Driessen, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, pp. 212–223, Aug. 1999.

[49] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 2000, pp. 93–104.

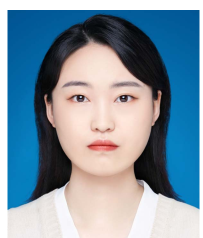
[50] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[51] C. A. Sims, "Macroeconomics and reality," *Econometrica, J. Econ. Soc.*, vol. 48, pp. 1–48, Jan. 1980.

[52] J. Hamilton, *Time Series Analysis*. Princeton, NJ, USA: Princeton Univ. Press, 2020. [Online]. Available: https://books.google.com.hk/books?id=BeryDwAAQBAJ

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

[54] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.



Leyan Deng received the B.S. degree in mathematics from Shandong University, Jinan, China, in 2019. She is currently pursuing the M.E. degree with the School of Data Science, University of Science and Technology of China, Hefei, China.

Her current research interests include anomaly detection and spatiotemporal data mining.

Defu Lian received the Ph.D. degree in computer science from the University of Science and Technology of China (USTC), Hefei, China, in 2014.

He is currently a Professor with the School of Data Science, USTC. He has published prolifically in refereed journals and conference proceedings, such as *ACM Transactions on Intelligent Systems and Technology*, *ACM Transactions on Information Systems*, IEEE TRANSACTION ON KNOWLEDGE AND DATA ENGINEERING, AAAI Conference on Artificial Intelligence (AAAI) Conference on Artificial Intelligence, IEEE International Conference on Data Mining (ICDM), ACM SIGKDD Conference on Knowledge Discovery and Data Mining, ACM International Conference on Research on Development in Information Retrieval, International Joint Conferences on Artificial Intelligence, and ACM International World Wide Web Conferences. His current research interests include spatial data mining, recommender systems, and learning to hash.

Zhenya Huang received the Ph.D. degree in computer science from the University of Science and Technology of China (USTC), Hefei, China, in 2020.

He is currently an Associate Researcher with the School of Computer Science and Technology, USTC. He has published several papers in referred journals and conference proceedings, such as *ACM Transactions on Information Systems*, AAAI Conference on Artificial Intelligence, ACM International Conference on Research on Development in Information Retrieval, ACM International Conference on Information and Knowledge Management, International Conference on Database Systems for Advanced Applications, and ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD). His current research interests include recommender systems, intelligent tutoring systems, educational data analysis, and other related topics in data mining, representation learning, and artificial intelligence.

Enhong Chen (Senior Member, IEEE) received the Ph.D. degree in computer science from the University of Science and Technology of China (USTC), Hefei, China, in 1996.

He is currently a Professor and the Vice Dean of the School of Data Science, USTC. He has published more than 200 papers in refereed conferences and journals, including the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON MOBILE COMPUTING, ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), IEEE International Conference on Data Mining (ICDM), Conference on Neural Information Processing Systems, and ACM International Conference on Information and Knowledge Management. His current research interests include data mining and machine learning, social network analysis, and recommender systems.

Dr. Chen was a recipient of the Best Application Paper Award on KDD 2008, the Best Research Paper Award on ICDM 2011, and the Best of SIAM International Conference on Data Mining (SDM) 2015. He was on the program committee of numerous conferences, including KDD, ICDM, and SDM.