



Article

Self-Supervised Spatiotemporal Masking Strategy-Based Models for Traffic Flow Forecasting

Gang Liu ¹, Silu He ^{2,*}, Xing Han ², Qinyao Luo ², Ronghua Du ³, Xinsha Fu ⁴ and Ling Zhao ²¹ China Academy of Electronic Information Technology, Beijing 100041, China; liugang10@cetc.com.cn² School of Geosciences and Info-Physics, Central South University, Changsha 410083, China; martinhe@csu.edu.cn (X.H.); qinyaoluo@csu.edu.cn (Q.L.); zhaoling@csu.edu.cn (L.Z.)³ College of Automotive and Mechanical Engineering, Changsha University of Science and Technology, Changsha 410114, China; csdrh@163.com⁴ School of Civil Engineering and Transportation, South China University of Technology, Guangzhou 510640, China; fuxinsha@163.com

* Correspondence: hesilu@csu.edu.cn

Abstract: Traffic flow forecasting is an important function of intelligent transportation systems. With the rise of deep learning, building traffic flow prediction models based on deep neural networks has become a current research hotspot. Most of the current traffic flow prediction methods are designed from the perspective of model architectures, using only the traffic features of future moments as supervision signals to guide the models to learn the spatiotemporal dependence in traffic flow. However, traffic flow data themselves contain rich spatiotemporal features, and it is feasible to obtain additional self-supervised signals from the data to assist the model to further explore the underlying spatiotemporal dependence. Therefore, we propose a self-supervised traffic flow prediction method based on a spatiotemporal masking strategy. A framework consisting of symmetric backbone models with asymmetric task heads were applied to learn both prediction and spatiotemporal context features. Specifically, a spatiotemporal context mask reconstruction task was designed to force the model to reconstruct the masked features via spatiotemporal context information, so as to assist the model to better understand the spatiotemporal contextual associations in the data. In order to avoid the model simply making inferences based on the local smoothness in the data without truly learning the spatiotemporal dependence, we performed a temporal shift operation on the features to be reconstructed. The experimental results showed that the model based on the spatiotemporal context masking strategy achieved an average prediction performance improvement of 1.56% and a maximum of 7.72% for longer prediction horizons of more than 30 min compared with the backbone models.

Keywords: self-supervised learning; spatiotemporal contextual association; spatiotemporal graph neural networks; traffic flow forecasting



check for updates

Citation: Liu, G.; He, S.; Han, X.; Luo, Q.; Du, R.; Fu, X.; Zhao, L. Self-Supervised Spatiotemporal Masking Strategy-Based Models for Traffic Flow Forecasting. *Symmetry* **2023**, *15*, 2002. <https://doi.org/10.3390/sym15112002>

Academic Editor: Christos Volos

Received: 3 October 2023

Revised: 20 October 2023

Accepted: 25 October 2023

Published: 31 October 2023



Copyright: © 2023 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

Traffic flow forecasting is an important research topic in the field of intelligent transportation systems [1]. Accurate traffic forecasting is of great practical importance for traffic management and planning, alleviating traffic congestion phenomena, improving travel efficiency, and ensuring safe transportation [2,3]. Traffic flow prediction faces the challenge of adequately capturing the complex spatiotemporal dependence in traffic data. As shown in Figure 1, traffic flow data can be regarded as a form of spatiotemporal sequential data, which contain complex spatial, temporal, and spatiotemporal correlations. Since the traffic system is a complex system with numerous factors affecting the traffic state, it is difficult to fully and completely describe the spatiotemporal relationships among various factors in the traffic system using conventional mathematical methods.

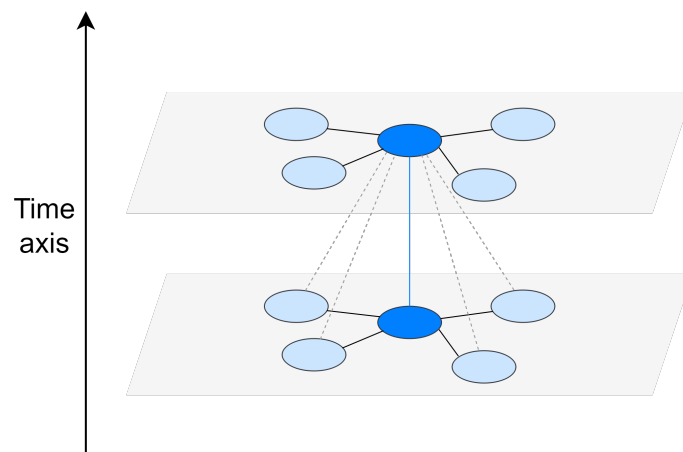


Figure 1. Spatiotemporal dependence in traffic flow data. Solid black lines indicate spatial dependence, solid blue lines indicate temporal dependence, and dotted lines indicate spatiotemporal dependence.

In recent years, as the collection of traffic data has become easier and the field of deep learning has rapidly developed, many data-driven, deep learning-based traffic flow prediction methods have emerged [4]. The core objective of such methods is to utilize deep neural networks to model the spatiotemporal dependencies in traffic data. The most common way is to adopt convolutional neural networks (CNNs) or graph neural networks (GNNs) to model spatial dependence in traffic data, and then combine them with temporal structures such as recurrent neural networks (RNNs) or one-dimensional convolution layers, and use the traffic flow features at future moments as supervision signals to guide the training process of the model in a data-driven way [5–11]. Among them, spatiotemporal graph neural networks (STGNNs) integrating GNNs with various temporal learning methods have shown promise in jointly modeling the spatial and temporal dependence, and have become the state-of-the-art architecture in traffic flow forecasting. Various GNN modules have been designed to capture the spatial correlation in traffic data. MRA-BGCNs [12] leverage the multi-range attention mechanism with the bi-component graph convolution to better aggregate information from different neighbors. STMGCNs [13] adopt multi-graph convolution to characterize the different spatial dependence. DMGCRNs [14] utilize hyperbolic GNNs to capture multi-scale spatial relationships. Different graph structures that can model spatial correlations from different perspectives have also been proposed, including pre-defined spatial graphs, spatial-temporal fusion graphs [15], localized spatial-temporal graphs [16], causality graphs [17], and adaptive multi-level fusion graphs [18].

However, traffic flow data themselves contain rich spatiotemporal features, and it is feasible to construct self-supervised signals from the data to assist the model in exploring the spatiotemporal dependence, for example, by forcing the model to distinguish the spatiotemporal samples generated from the original data to discriminate different spatiotemporal features [19,20], or by pre-training an effective transformer through the masked auto-encoding strategy to learn segment-level representations [21]. Therefore, the potential of STGNNs to perform traffic flow prediction tasks in a self-supervised manner is yet to be further explored. In this paper, based on existing spatiotemporal graph neural networks (STGNNs), we propose a spatiotemporal context mask reconstruction task, forcing the model to reconstruct the masked traffic features by utilizing spatiotemporal contextual information to better understand the spatiotemporal contextual associations in the data. With the assistance of the self-learning auxiliary task, the backbone model can achieve better performance on the primary prediction task [22]. The major contributions of this paper can be summarized as follows:

1. We propose a spatiotemporal context mask reconstruction task to force the model to reconstruct the masked traffic features based on the spatiotemporal contextual information, so as to enhance the existing STGNNs' understanding of spatiotemporal contextual associations and improve their prediction capability;
2. A specific spatiotemporal masking strategy is proposed to assist the model in understanding the spatiotemporal associations of each local part of the traffic network, and the effects of different masking strategies and masking ratios on the model performance are compared comprehensively;
3. We validated the proposed method on two real-world traffic datasets, and the experimental results show that introducing the spatiotemporal context mask reconstruction task as an auxiliary task can improve the prediction performance of STGNNs under the prediction horizons of 30, 45, and 60 min.

2. Related Work

2.1. Spatial Modeling

Most of the early traffic flow prediction methods based on deep neural networks only migrated the models designed for multi-variate time series prediction, ignoring the characteristics that exist in the traffic flow data themselves and lacking the modeling of the spatial relations in the road network [23–25]. With the successful application of convolutional neural network (CNN) models on image data, some of the approaches use CNNs to model the spatial relations of traffic data organized as a grid [11,26–29]. Shi et al. were the first to combine two-dimensional convolutional networks applied to images with LSTM [30] to capture both temporal and spatial dependence in the data, and this model was subsequently applied to the traffic flow [26]. Ma et al. [27] converted road network-based traffic flow data into a spatiotemporal matrix representation and then used a CNN to extract features from the spatiotemporal matrix.

CNN-based traffic flow forecasting methods are more concerned with the Euclidean distance and spatial location relations between roads, and lack the consideration of road topological relations. However, a road network is a natural graph structure in which the spatial topological relations are more suitable to be described in the form of graphs, and information such as the distance between roads can be represented in the form of the weights of edges. Inspired by CNNs and graph embedding methods, researchers have proposed a series of graph neural network (GNN) methods to aggregate information based on the graph structure. GNNs can be broadly classified into two categories, namely, spectral graph neural networks and spatial graph neural networks [31]. Bruna et al. proposed a spectral convolutional neural network [32], which generalized CNNs to non-Euclidean spaces, but the spectral decomposition process has a very high computational complexity. The spatial graph neural networks describe the graph convolution as an aggregation operation of nodes with information from their neighbors. DCNNs [33] simulate the diffusion process by considering the aggregation of information from nodes with their multi-order neighboring nodes. A GAT [34], on the other hand, dynamically learns the weights of information from neighboring nodes during aggregation through an attention mechanism and is able to better focus on the information of important nodes. Self-attention networks can also be adopted to extract spatial patterns dynamically [35,36]. With the capability of modeling data with irregular structures, GNNs can solve the problem that previous deep learning methods have, i.e., difficulty in modeling non-Euclidean data structures and capturing complex spatial dependence [37]. Therefore, most of the mainstream deep learning-based traffic flow forecasting methods currently use GNNs to learn the spatial dependence in traffic systems.

2.2. Temporal Modeling

Based on GNNs, researchers have further introduced the modeling of temporal dependence in traffic flow and proposed numerous spatiotemporal graph neural network (STGNN) models for traffic flow forecasting tasks, which have become the mainstream

traffic flow prediction models nowadays. Based on the modeling of temporal dependence, we further classify STGNNs into RNN-based methods, one-dimensional convolution-based methods, and self-attention mechanism-based methods.

1. RNN-based STGNNs. RNN-based STGNNs basically use a chain structure that combines a graph convolution module and a recurrent unit. DCRNNs [5] adopt a variant of an RNN, the gated recurrent unit (GRU), for the extraction of temporal features. The GRU is able to achieve a similar performance as LSTM with fewer parameters, which can effectively reduce the number of parameters and the training time [38]. T-GCNs [7], on the other hand, combine GCNs and the GRU and test them on traffic datasets with two common scenarios, namely, highways and urban roads, and obtain prediction performance that exceeds the baselines. However, due to the limitation of the chain structure of RNNs, the input of the subsequent time step depends on the output of the preceding time step and, thus, does not allow for parallel training of parameters;
2. One-dimensional convolution-based STGNNs. STGCNs [6] apply one-dimensional causal convolution and gated linear units to extract temporal features from traffic flow. Graph WaveNet [10] performs one-dimensional dilated causal convolution on temporal features, which makes the reception field of the model grow exponentially, thus facilitating the model to better capture the long-range temporal dependence in the data. One-dimensional convolution-based models are more computationally efficient compared to RNN-based models for modeling temporal dependence and also avoid the problem of gradient vanishing or explosion;
3. Self-attention mechanism-based STGNNs. The traffic transformer [8] designs various positional encoding strategies to learn the periodic features in traffic flow. STTNs [39] incorporate the graph convolution process into the spatial transformer, builds a spatiotemporal transformer block with the spatial and temporal transformer, and stacks the blocks to capture the dynamic spatiotemporal correlations in the traffic data. The self-attention mechanism ensures that the model parameters can be trained in parallel while making direct connections between the various time steps of the input sequence, which can help the self-attention-based traffic flow prediction models to better capture the long-range temporal dependence in traffic data.

2.3. Learning Paradigms

In addition to designing optimal architectures, an increasing number of advanced training paradigms have been applied in traffic flow prediction. The traditional supervised learning paradigm only uses labels as supervision signals to guide the training of the model. Commonly used loss functions such as L1 and L2 norms can guide the optimization of models, but may overlook the distribution underlying predicted data and real data. To cope with this problem, generative frameworks such as generative adversarial networks (GANs) and auto-encoders are applied in traffic prediction [40,41]. These methods can be defined as unsupervised models that integrate supervised loss as a part of learning process of STGNNs. For example, TFGANs [42] combine adversarial loss with an STGNN to ensure the distribution similarity between the predicted result and input data. GCGANs [43] use a GAN that combines graphic comparison learning.

In the era of big data, the amount of data available for use by deep learning models has grown dramatically and the demand for further exploration of the information embedded in the data is increasingly significant. How to learn the features of the data from the information provided by the data themselves has become a key research problem [44]. Self-supervised learning moves the unsupervised learning forward one step by creating inherent labels through self-supervised task. To assist the downstream prediction task, various self-supervised tasks are designed. STEP [21] pre-trained an effective transformer through the masked auto-encoding strategy to learn segment-level representations, which can help the downstream STGNN perform better at the prediction task. Banville et al. [45] proposed two auxiliary tasks to learn underlying temporal features in time series. Specifically, the relative

positioning task requires the model to determine the relative position of one window of a time series to another, while the ranking task requires the model to infer whether a given sequence has been randomly disrupted. STGCL [20] uses multiple sample augmentation methods in which positive and negative sample pairs are constructed to guide the STGNNs to distinguish different spatiotemporal features in samples.

3. Methodology

3.1. Overview

The overview of the proposed method is shown in Figure 2. The method can be divided into two branches, namely, the prediction branch (the upper pipeline in Figure 2) and the spatiotemporal context mask reconstruction branch (the lower pipeline in Figure 2).

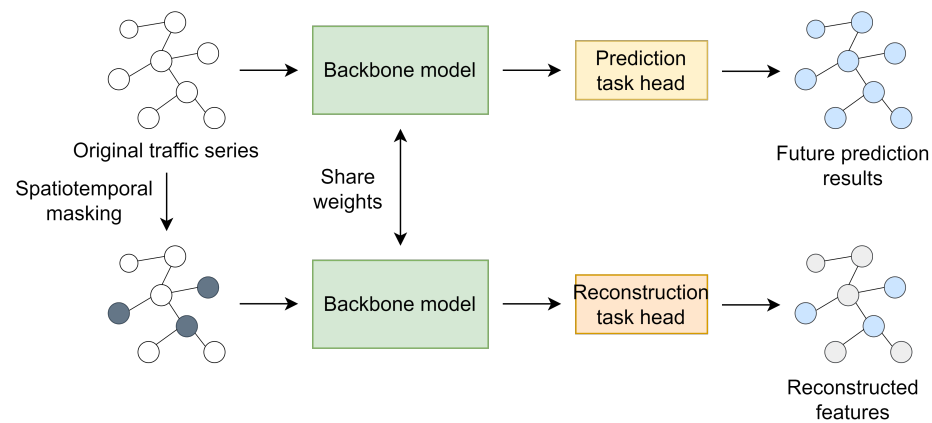


Figure 2. The overview of the spatiotemporal masking strategy-based traffic flow forecasting method.

Prediction branch. The prediction branch is the same as the previous supervised learning paradigm, and its main purpose is to use the traffic flow state at the future moment as the supervision signal for the model, and to guide the model to give prediction results that are as close as possible to the ground truth. In the prediction branch, the input of the backbone model is the original traffic flow sequence data X . The hidden representations are passed to the prediction task head consisting of MLPs to obtain the prediction results for the future moments.

Spatiotemporal context mask reconstruction branch. The spatiotemporal context mask reconstruction branch aims to utilize the self-learning auxiliary task to enhance the model's understanding of the underlying spatiotemporal dependence in the traffic flow. With this purpose, some of the traffic features in X are masked to force the model to reconstruct the corresponding features based on the spatiotemporal context information. In the auxiliary task branch, the original traffic flow sequence data are processed by specific spatiotemporal sampling and the masking strategy to obtain the augmented historical traffic flow sequence, which is used as the input of the backbone model, and the hidden representations are passed to the reconstruction task head composed of multi-layer perceptrons (MLPs) to obtain the reconstruction results with a temporal shift. The weights of the backbone models are shared with each other in both branches. The detailed spatiotemporal masking strategy will be described in the next subsection.

3.2. Spatiotemporal Context Masking

The spatiotemporal context masking strategy is shown in Figure 3. The strategy can be divided into two parts, i.e., the spatial masking strategy and the temporal masking strategy. The temporal masking strategy will be applied to the nodes that are sampled by the spatial masking strategy.

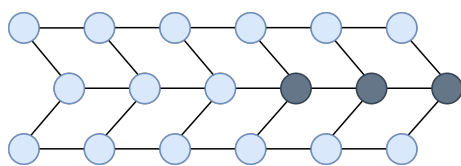


Figure 3. Spatiotemporal context masking. Each circle represents a feature point, the horizontal axis represents the time axis, and the vertical axis represents the spatial axis. Dark color indicates masked feature points.

Spatial masking strategy. The spatial masking strategy is implemented to learn spatial features that capture spatial dependence. When considering how to sample the nodes to be masked spatially, we should avoid sampling nodes that obey specific spatial distributions. This can lead to the model excessively focusing on a particular aspect of spatial dependence, thereby failing to capture other aspects, ultimately resulting in a decrease in the model's predictive performance at the level of the whole traffic network. In order to make the model fully learn the spatial dependence in each local part of the whole traffic network, a certain percentage of nodes is randomly sampled spatially for each batch of data. In addition, from the perspective of computational efficiency, we shared the random sampling results among the samples of the same batch, i.e., the nodes sampled in the same batch are the same.

Temporal masking strategy. The temporal masking strategy is implemented to learn temporal features that capture temporal dependence. Unlike the spatial masking strategy, if random discrete feature points are sampled as the masked features, the model will be prone to directly inferring the features of the masked discrete points from the features of the previous or next moment, and cannot genuinely infer the values of the masked features based on spatiotemporal dependence. Therefore, we consider continuous masking of temporal features on the time axis. In addition, considering our masking strategy is designed to assist the temporal prediction task, whose goal is to use the information from previous moments to predict the future moments, we chose to mask the latter part of the traffic features of the sampled nodes. This temporal masking strategy can further assist the model in learning the association between the previous moments and the future moments.

3.3. Temporal Shift

The input historical sequences of the model are short-term series. Compared with long-term series, the information redundancy of short-term series is low; in addition, a large proportion of masked temporal features will lead to serious information loss, and the model cannot understand the temporal dependence and reconstruct temporal features from the extremely limited temporal context; thus, a small mask ratio was chosen in this paper. However, when the mask ratio is small, the model can easily make simple inferences to reconstruct the masked temporal features based on the local smoothness in the data [46,47], which cannot help the model to truly understand the spatiotemporal correlations. Moreover, as mentioned in the previous subsection, this paper considers only one downstream task, i.e., the prediction task, and we should assist the model in learning the association between previous moments and future moments. Therefore, we shifted the temporal features to be reconstructed, i.e., masking the temporal features x_t in the moment t and requiring the model to reconstruct the temporal features $x_{t \rightarrow s}$ in the moment $t + s$, where s is the temporal shift, to avoid the model from reconstructing features through shortcuts while still ensuring that the model reconstructs the future moments based on the previous moments. When $s = 0$, the auxiliary task regresses to the original mask reconstruction task.

3.4. Loss Function

There are two tasks in the training stage of the model, namely, the prediction task and the spatiotemporal context mask reconstruction task. The prediction task adopts the L1 loss function so that the predictions given by the model are as similar as possible to the actual traffic features:

$$\mathcal{L}_{\text{forecast}}(\hat{\mathbf{Y}}, \mathbf{Y}; \Theta) = \|\hat{\mathbf{Y}} - \mathbf{Y}\| \quad (1)$$

where $\hat{\mathbf{Y}}$ denotes the predictions given by the model, \mathbf{Y} denotes the ground truth, and Θ denotes the set of all learnable parameters of the model.

The spatiotemporal context mask reconstruction task also uses the L1 loss function so that the reconstruction result $\hat{\mathbf{X}}_{\text{masked}}^{\rightarrow s}$ is as consistent as possible with the masked traffic features after the temporal shift $\mathbf{X}_{\text{masked}}^{\rightarrow s}$. Only the masked features are considered when computing loss, and the unmasked features are not involved in the calculation of the loss function:

$$\mathcal{L}_{\text{reconstruct}}(\hat{\mathbf{X}}_{\text{masked}}^{\rightarrow s}, \mathbf{X}_{\text{masked}}^{\rightarrow s}; \Theta) = \|\hat{\mathbf{X}}_{\text{masked}}^{\rightarrow s} - \mathbf{X}_{\text{masked}}^{\rightarrow s}\| \quad (2)$$

Following the auxiliary learning paradigm that the self-learning task serves as the assistant of the primary prediction task, the model is trained with the complete loss function, consisting of the loss of forecasting task and weighted loss of reconstructing task:

$$\mathcal{L} = \mathcal{L}_{\text{forecast}} + \lambda \mathcal{L}_{\text{reconstruct}} \quad (3)$$

where λ denotes the weight of the loss function of the auxiliary task.

4. Experiments

4.1. Datasets

We conducted experiments on two real-world datasets: METR-LA and PEMS-BAY, and the basic information of the datasets is shown in Table 1. The datasets were split with 70% for training, 10% for validation, and 20% for testing.

1. METR-LA: Traffic speed dataset. It comprises 4 months of data from the highway of Los Angeles with the temporal range of 2012/3/1–2012/6/30;
2. PEMS-BAY: Traffic speed dataset. It comprises 6 months of data from the Bay Area with the temporal range of 2017/1/1–2017/6/30.

Table 1. The basic statistics of two real-world traffic state datasets.

Dataset	#Nodes	#Edges	Sparsity	Sampling Interval	#Sampling Points
METR-LA	207	1722	4.02%	5 min	34,272
PEMS-BAY	325	2694	2.55%	5 min	52,116

The spatial distributions of nodes in the datasets are shown in Figures 4 and 5.

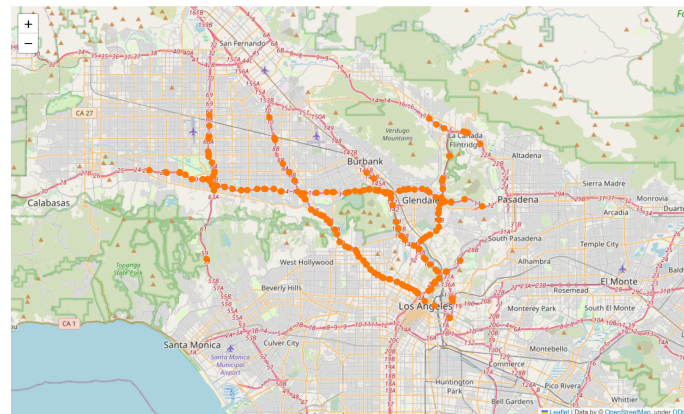


Figure 4. The spatial distribution of nodes in the METR-LA dataset. Traffic nodes are marked in orange.

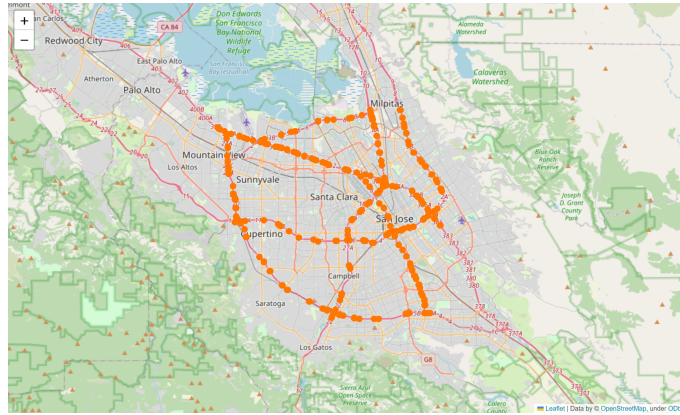


Figure 5. The spatial distribution of nodes in the PEMS-BAY dataset. Traffic nodes are marked in orange.

4.2. Evaluation Metrics

Let the number of samples in the dataset be n , $\mathbf{Y}_i = \mathbf{X}_{t:t+F} = [\mathbf{X}_t, \mathbf{X}_{t+1}, \dots, \mathbf{X}_{t+F-1}]$ denotes the traffic features to be predicted of the i -th sample in the dataset, and $\hat{\mathbf{Y}}_i$ denotes the prediction result given by the model. The following metrics are used to measure the prediction performance of the model:

1. Root mean squared error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{\mathbf{Y}}_i - \mathbf{Y}_i)^2} \quad (4)$$

2. Mean absolute error (MAE)

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |\hat{\mathbf{Y}}_i - \mathbf{Y}_i| \quad (5)$$

3. Mean absolute percentage error (MAPE)

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{\hat{\mathbf{Y}}_i - \mathbf{Y}_i}{\mathbf{Y}_i} \right| \times 100\% \quad (6)$$

For all of the above metrics, lower values indicate smaller errors and more accurate predictions. Among them, MAE provides a direct measurement of prediction residual and MAPE represents the average magnitude of relative error in the dataset; while a higher RMSE can emphasize larger errors in the sample data.

4.3. Backbone Models and Hyperparameter Settings

In order to comprehensively evaluate the effect of the proposed method, two STGNNs with different structures (i.e., RNN-based and 1D convolution-based) were selected as the backbone models. In order to best ensure the predictive performance of backbone models and utilize them as a comparative baseline, we adopted the default settings for the model architecture and hyperparameters as described in their papers [7,10]. The descriptions and hyperparameter settings were as follows:

1. A T-GCN [7] is an STGNN that can be regarded as an RNN-based model and combines a GCN with a GRU. For the T-GCN, the batch size was set to 32, and the number of hidden units was set to 64. Adam [48] was chosen as the optimizer, and the learning rate was set to 0.001;
2. Graph WaveNet [10] is an STGNN that can be seen as a 1D convolution-based model, which uses adaptive graph convolution to capture spatial dependence and 1D con-

volution to capture temporal dependence. For Graph WaveNet, the batch size was set to 32, and the probability of dropout in the graph convolution layer was set to 0.3. Adam [48] was chosen as the optimizer, and the learning rate was set to 0.001.

Other hyperparameters were set as follows: the number of MLP layers for both the prediction task head and the reconstruction task head was 2, and the hidden layer dimension of the MLPs was set to 32. Taking into account the periodicity of traffic flow data, the traffic features at an interval of 12 time steps can exhibit both periodicity and dynamic changes. In addition to the consideration of the maximum horizon of the multi-step prediction evaluated in this paper, we set the temporal shift s to 12.

4.4. Experimental Results and Analysis

4.4.1. Accuracy

In order to keep the number of parameters of the prediction task head and the auxiliary task head consistent, the backbone structures of the STGNNs were maintained and connected to prediction task heads; the models connected to prediction task heads and reconstruction task heads and trained with the auxiliary spatiotemporal context mask reconstruction task were named the “STC-backbone model”, and the experimental results are shown in Tables 2 and 3.

Table 2. The prediction performance of spatiotemporal context masking-based models and original backbone models on the METR-LA dataset.

Prediction Horizon	Evaluation Metrics	T-GCN	STC-T-GCN	Graph WaveNet	STC-Graph WaveNet
15 min	RMSE	5.11	5.10	4.85	4.85
	MAE	2.63	2.64	2.60	2.63
	MAPE	6.99%	6.97%	6.64%	6.75%
30 min	RMSE	5.95	5.91	5.98	5.84
	MAE	2.99	2.98	3.09	3.04
	MAPE	8.24%	8.11%	8.80%	8.37%
45 min	RMSE	6.48	6.45	6.67	6.58
	MAE	3.33	3.32	3.40	3.34
	MAPE	9.11%	9.05%	10.03%	9.60%
60 min	RMSE	6.88	6.84	7.63	7.35
	MAE	3.41	3.40	3.87	3.79
	MAPE	9.48%	9.45%	11.79%	10.88%

Table 3. The prediction performance of spatiotemporal context masking-based models and original backbone models on the PEMS-BAY dataset.

Prediction Horizon	Evaluation Metrics	T-GCN	STC-T-GCN	Graph WaveNet	STC-Graph WaveNet
15 min	RMSE	2.48	2.48	2.47	2.49
	MAE	1.25	1.25	1.17	1.18
	MAPE	2.57%	2.58%	2.34%	2.44%
30 min	RMSE	3.17	3.14	3.47	3.42
	MAE	1.49	1.48	1.52	1.52
	MAPE	3.26%	3.24%	3.40%	3.35%
45 min	RMSE	3.67	3.65	4.18	4.15
	MAE	1.67	1.66	1.82	1.81
	MAPE	3.81%	3.77%	4.06%	4.25%
60 min	RMSE	3.93	3.91	4.90	4.68
	MAE	1.79	1.78	2.08	2.05
	MAPE	4.14%	4.08%	5.20%	4.97%

As shown in Table 2, on the METR-LA dataset, with the spatiotemporal context mask reconstruction task, the models showed an improvement in prediction under 30, 45, and 60 min horizons compared with the corresponding backbone models. Compared with the T-GCN, the STC-T-GCN decreased the RMSE metrics by 0.67%, 0.46%, and 0.58%, the MAE metrics by 0.33%, 0.30%, and 0.29%, and the MAPE metrics by 1.58%, 0.66%, and 0.32% under 30 min, 45 min, and 60 min horizons, respectively. The spatiotemporal context mask reconstruction task had a more obvious effect on the prediction performance improvement of Graph WaveNet. Compared with Graph WaveNet, the RMSE metric of STC-Graph WaveNet decreased by 2.34%, 1.35%, and 3.67% at 30, 45, and 60 min horizons, respectively, the MAE metric decreased by 1.62%, 1.76%, and 2.07%, respectively, and the MAPE metrics decreased by 4.89%, 4.29%, and 7.72%, respectively. It can be concluded that the spatiotemporal context mask reconstruction task can improve the prediction accuracy of models for prediction horizons of 30 min and above on the METR-LA dataset.

As shown in Table 3, with the spatiotemporal context mask reconstruction task, the models still performed better compared with the backbone models for horizons of 30 min or more on the PEMS-BAY dataset. Compared with the T-GCN, the STC-T-GCN decreased the RMSE metrics by 0.95%, 0.67%, and 0.51%, the MAE metrics by 0.67%, 0.60%, and 0.56%, and the MAPE metrics by 0.61%, 1.05%, and 1.45% for the 30, 45, and 60 min horizons, respectively. Compared with Graph WaveNet, STC-Graph WaveNet's RMSE metrics decreased by 1.44%, 0.72%, and 4.49% for the 30 min, 45 min, and 60 min forecast horizons, respectively; the MAE metrics decreased by 0.55% and 1.44% for the 45 min and 60 min horizons, respectively; the MAPE metric decreased by 1.47% and 4.42% under the 30 min and 60 min forecast horizons, respectively. In summary, the spatiotemporal context mask reconstruction task also had a certain enhancement effect on the prediction performance of the models on the PEMS-BAY dataset.

Considering the performance of the models on the two datasets, it can be seen that the performance of the models with the spatiotemporal context mask reconstruction task improved at the horizons of 30, 45, and 60 min, while there was little or no improvement at the prediction length of 15 min. This may be because, for short-term predictions, the temporal features of each node play a more important role in prediction, while the underlying spatiotemporal dependence information in the data is not obvious in this situation; while for more complex long-term prediction tasks, the spatiotemporal dependence in the data plays a more important role, and a better understanding of the spatiotemporal dependence can help the model to better accomplish the prediction task.

4.4.2. Masking Strategies

In order to verify its effectiveness, we compared the spatiotemporal context masking strategy with the following four strategies:

1. Degree centrality masking. Nodes with a higher degree centrality may play more important roles in the traffic network. Compared to the spatiotemporal context masking strategy, this strategy replaces the random node sampling with degree centrality-based node sampling;
2. Spatial masking. As shown in Figure 6a, after randomly sampling nodes, all the temporal features of the selected nodes are masked;
3. Temporal masking. As shown in Figure 6b, this strategy masks the temporal features of all nodes near the current moment;
4. Completely random masking. As shown in Figure 6c, this strategy randomly masks a fixed proportion of traffic feature points in the whole traffic feature matrix.

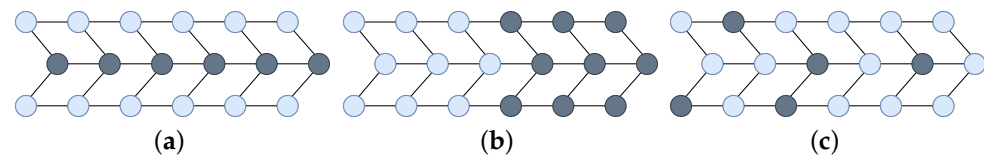


Figure 6. Masking strategies for comparison. Dark color indicates masked feature points. (a) Spatial masking, (b) temporal masking, and (c) complete random masking. Each circle represents a feature point, the horizontal axis represents the time axis, and the vertical axis represents the spatial axis.

In this subsection, we describe analyses in which Graph WaveNet was adopted as the backbone model, and the experiments were conducted on the METR-LA dataset at a horizon of 60 min. The prediction performance of the models with different masking strategies is shown in Table 4. From the table, it can be seen that the fixed sampling of nodes with high centrality caused the model to focus excessively on the spatiotemporal dependence of these nodes, which negatively affected the prediction performance at the overall traffic network level. Compared with the spatiotemporal context masking strategy, the RMSE, the MAE, and the MAPE metrics of the degree centrality masking strategy increased by 4.34%, 4.35%, and 3.58%, respectively. In addition, the spatial masking and the temporal masking strategies, which focus on one dimension alone, also underperformed, with the MAE metric increasing by 2.97%, indicating that the integrated consideration of the spatiotemporal context helped to improve the prediction of the model. Finally, the RMSE, the MAE, and the MAPE metrics of the completely random masking strategy increased by 2.72%, 2.97%, and 2.39%, respectively. This may be because the masked features were more dispersed and easier to be directly inferred by interpolation, and the difficulty of the auxiliary task was lower, and so could not play a substantial role in assisting the model to understand the spatiotemporal dependence in the data.

Table 4. The prediction performance of models with different masking strategies.

	RMSE	MAE	MAPE
Spatiotemporal context masking	7.35	3.79	10.88%
Degree centrality masking	7.67	3.85	11.27%
Spatial masking	7.52	3.81	11.59%
Temporal masking	7.45	3.81	11.53%
Completely random masking	7.55	3.81	11.14%

4.4.3. Hyperparameter Analysis

There are two main hyperparameters in this method, i.e., the ratio of masked nodes and the weight of the auxiliary task in the loss function λ . We analyzed the settings of these two parameters using Graph WaveNet as the backbone model with a prediction horizon of 60 min on the METR-LA dataset.

(1) The ratio of masked nodes.

In order to find the best ratio of the masked nodes, the mask ratio was set to 10%, 20%, 30%, 50%, 70%, and 90%, and the experimental results are shown in Figure 7. From the figure, it can be seen that, as the mask ratio increased, the RMSE, the MAE, and the MAPE metrics showed a general trend of decreasing and then increasing, and the value of each metric reached the minimum when the mask ratio was 20%. A low mask ratio will reduce the difficulty of the auxiliary task and cannot fully help the model understand the spatiotemporal dependence in the data, while a high mask ratio will make the auxiliary task too difficult and will play a negative role in terms of the model completing the main prediction task. Since the model inputs in this paper are short-term historical series with relatively low information redundancy, an excessive mask ratio will cause serious information loss, making it difficult for the model to reconstruct features and explore spatiotemporal associations.

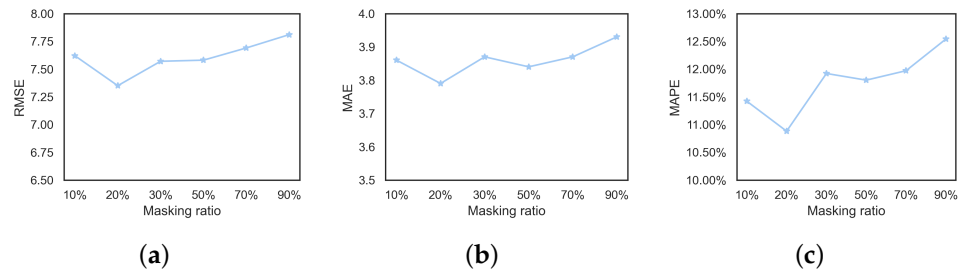


Figure 7. The prediction performance of models with different masking ratios. (a) RMSE, (b) MAE, and (c) MAPE.

(2) λ value in the loss function.

The value of λ of the loss function determines the weight of the auxiliary task. In order to find the best value of λ , we set λ as 0.02, 0.05, 0.1, 0.2, and 0.5, and conducted experiments. The experimental results are shown in Figure 8. From the figure, it can be seen that for the RMSE metric, the effect of λ was not obvious; while for MAE and MAPE, a more obvious trend was found. As the value of λ increased, the values of MAE and MAPE showed the characteristics of first decreasing and then increasing, and the lowest value was reached when the value of λ was 0.1. When the value of λ was too low, the role of the auxiliary task was not significant, so the enhancement effect was more limited; while when the value of λ was too high, the weight of the auxiliary task was larger, which affected the training of the prediction task and played a negative role in the prediction performance of the model.

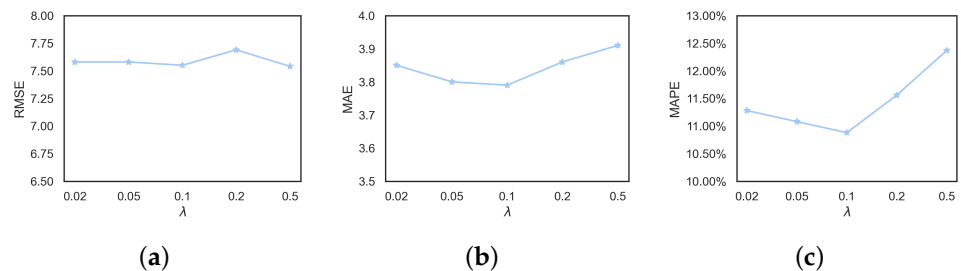


Figure 8. The prediction performance of models with different λ values. (a) RMSE, (b) MAE, and (c) MAPE.

4.4.4. Visualization

In order to visually compare the prediction performance of the backbone models and the models with the spatiotemporal context mask reconstruction task, we selected two representative nodes and visualized the prediction results of Graph WaveNet and STC-Graph WaveNet on them. As shown in Figure 9, sensor #717816 (i.e., the 9th node) was located downstream of the convergence of two segments in the same direction and #769373 (i.e., the 206th node) was located downstream of a segment's split. They both had a more complex spatial dependence, thus leading to a higher uncertainty in their traffic feature patterns and prediction difficulty because of the synchronous impact from two road segments. The prediction results of Graph WaveNet and STC-Graph WaveNet on sensor #717816 in the METR-LA dataset from 5 June to 12 June 2012 are visualized in this paper, and the prediction results for horizons of 15, 30, 45, and 60 min are shown in Figure 10, Figure 11, Figure 12, and Figure 13, respectively. The prediction results on sensor #769373 for horization of 15 min is shown in Figure 14.

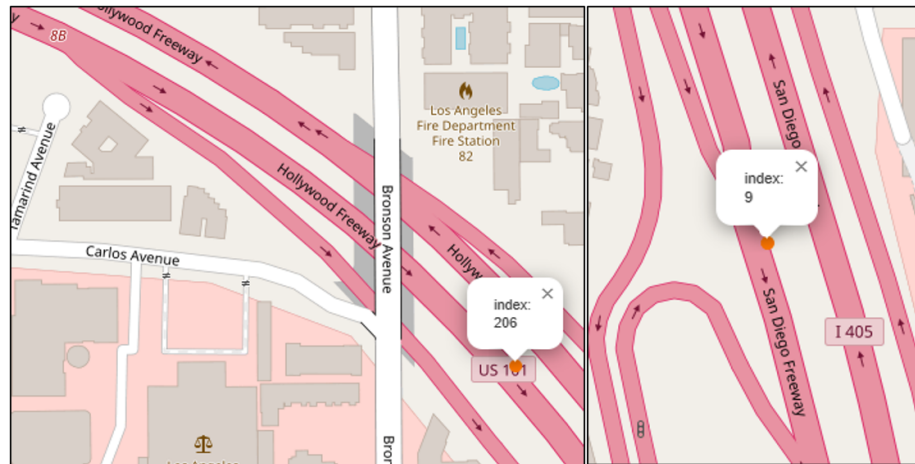


Figure 9. Location of visualized sensors.

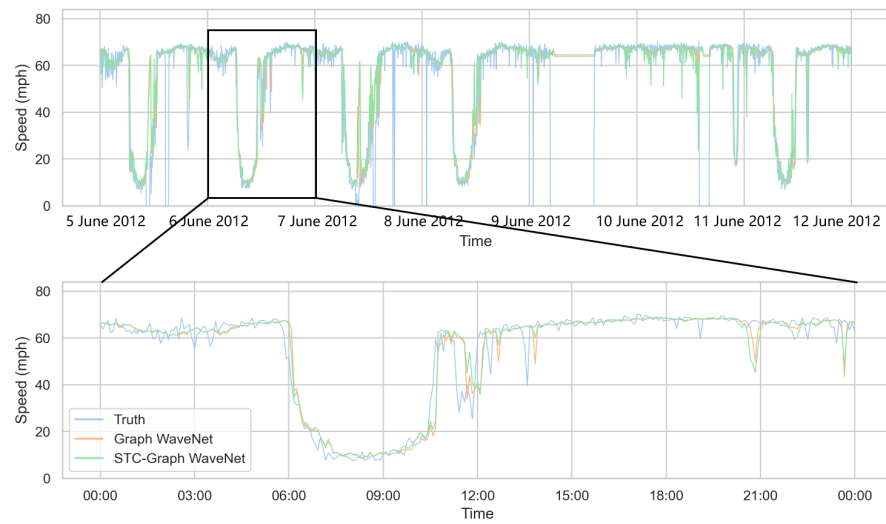


Figure 10. Visualization of predictions 15 min ahead on sensor #717816.

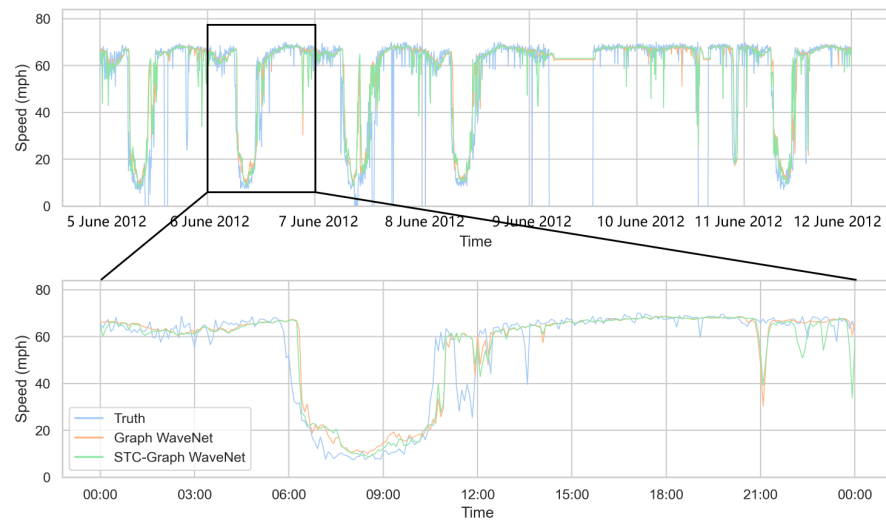


Figure 11. Visualization of predictions 30 min ahead on sensor #717816.

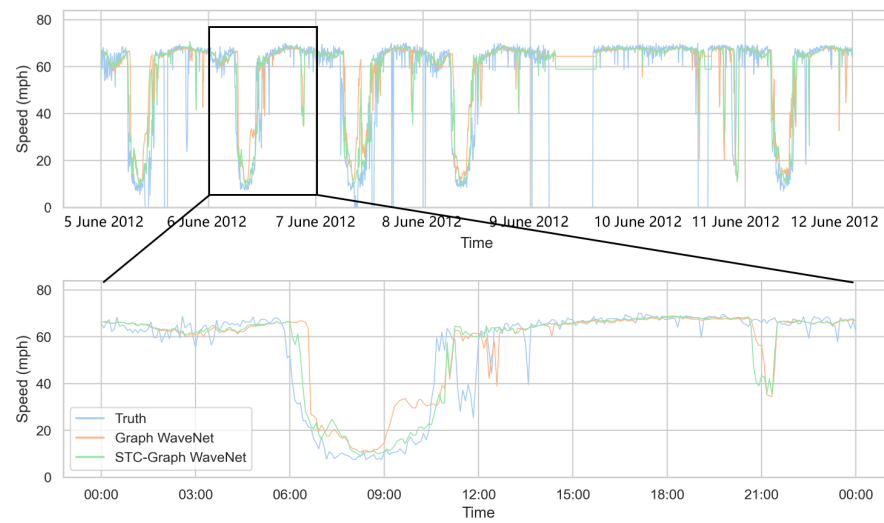


Figure 12. Visualization of predictions 45 min ahead on sensor #717816.

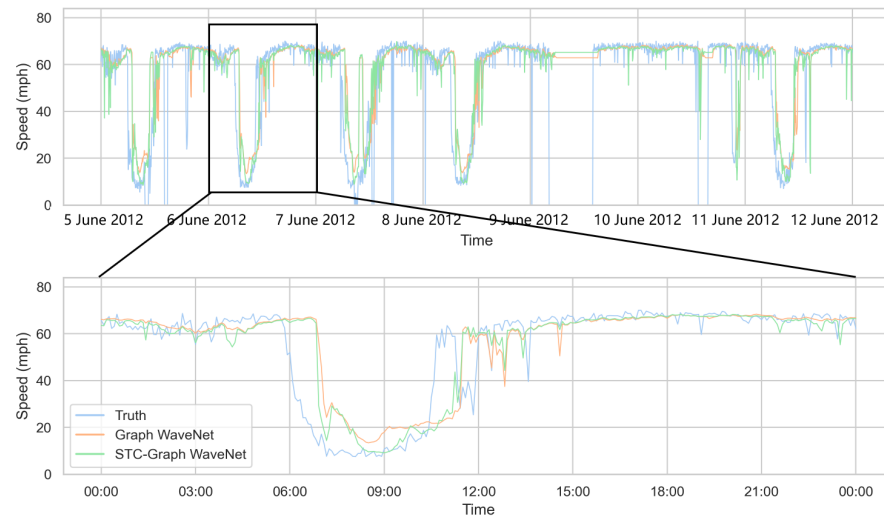


Figure 13. Visualization of predictions 60 min ahead on sensor #717816.



Figure 14. Visualization of predictions 15 min ahead on sensor #769373.

From the prediction results, it can be seen that Graph WaveNet as the backbone model had a good prediction performance and could predict the changes of traffic flow in future moments accurately. However, when facing some scenarios with large fluctuations in traffic features, the performance of Graph WaveNet was mediocre. For example, during the morning peak hours (06:00 to 10:30), the number of vehicles in the road network increased due to the commuting demand of residents, and the traffic speed detected by the sensors decreased significantly. It can be observed that the prediction accuracy of Graph WaveNet for traffic features in the morning peak hour gradually decreased as the prediction horizon increased, while STC-Graph WaveNet was able to better fit the morning peak hour traffic flow features. In addition, for sensor #769373 during highly dynamic phases such as morning and evening peak hours, STC-Graph WaveNet was capable of consistently generating short-term predictions that were closer to the truth (Figure 14). This indicates that the backbone model can achieve better understanding of complex spatiotemporal dependence by training with the auxiliary spatiotemporal mask reconstruction task.

In addition, both Graph WaveNet and STC-Graph WaveNet still had some scenarios where the predictions were not accurate enough. For example, the traffic speed around 21:00 showed a very smooth trend, while the prediction results given by both Graph WaveNet and STC-Graph WaveNet showed an abrupt speed drop and then an increase. This may be because the prediction results of the current node were influenced by the temporal features of other nodes in the road network, i.e., the model passed the temporal features present in other nodes at the current prediction moment to the current node through the message passing mechanism of the graph neural network. In the future, we will further consider how to assist the model in learning more accurate spatiotemporal causal dependence to avoid the interference of irrelevant temporal features.

5. Conclusions

In order to assist the model in better understanding the underlying spatiotemporal contextual associations in traffic flow data to improve the prediction capability of models, we designed a traffic flow prediction method based on a self-supervised spatiotemporal masking strategy. Firstly, the spatiotemporal context mask reconstruction task was used as an auxiliary task to provide additional self-supervised signals to guide the training process of the model; then a specific spatiotemporal context mask sampling strategy was designed for the prediction task from both the spatial and temporal perspectives; at the same time, in order to avoid the model finishing the auxiliary task based on the local smoothness in the data and failing to learn the real spatiotemporal correlations, we applied a temporal shift operation to the features to be reconstructed. Finally, we verified the effectiveness of our method through experiments using two backbone models on two datasets. In addition, we analyzed different masking strategies, important hyperparameters in the method, and the visualized prediction results. The following conclusions can be drawn:

1. Compared with backbone models, the models based on the self-supervised spatiotemporal masking strategy have a better prediction performance at horizons of 30, 45, and 60 min. The average prediction performance improvement reaches 1.56% at horizons of more than 30 min, which proves that the proposed method improves spatiotemporal dependence understanding of the model and can be helpful for long-term prediction;
2. Comparing different masking strategies, it was found that considering a single dimension, such as only the spatial dependence or only the temporal dependence, has a relatively limited improvement effect on the model performance, while considering both spatial and temporal perspectives together can more effectively improve the prediction capability;
3. The visualization results show that for scenarios with large fluctuations, the proposed method is able to give prediction results with a better fit to the actual values. However, sometimes the model is also affected by confounding spurious spatiotemporal correlations, leading to erroneous prediction results.

Author Contributions: Conceptualization, X.H. and S.H.; methodology, X.H. and G.L.; software, G.L.; validation, X.H. and S.H.; formal analysis, X.H., G.L. and S.H.; investigation, R.D. and Q.L.; resources, Q.L. and L.Z.; data curation, X.F.; writing—original draft preparation, X.H. and G.L.; writing—review and editing, X.H., G.L. and S.H.; visualization, X.F. and Q.L.; supervision, R.D. and L.Z.; project administration, S.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Major Program Project of Xiangjiang Laboratory under Grant 22XJ01010, and the National Natural Science Foundation of China, Grant/Award Numbers: 42301381 and 42271481; and in part by using Computing Resources at the High-Performance Computing Platform of Central South University.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Acknowledgments: We acknowledge the High-Performance Computing Platform of Central South University and HPC Central of the Department of GIS for providing HPC resources.

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

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional neural network
GNN	Graph neural network
RNN	Recurrent neural network
STGNN	Spatiotemporal graph neural network
GRU	Gated recurrent unit
MLP	Multi-layer perceptrons

References

1. Yuan, H.; Li, G. A survey of traffic prediction: From spatio-temporal data to intelligent transportation. *Data Sci. Eng.* **2021**, *6*, 63–85. [[CrossRef](#)]
2. Nagy, A.M.; Simon, V. Survey on traffic prediction in smart cities. *Pervasive Mob. Comput.* **2018**, *50*, 148–163. [[CrossRef](#)]
3. Hashemi, S.M.; Botez, R.M.; Grigorie, T.L. New Reliability Studies of Data-Driven Aircraft Trajectory Prediction. *Aerospace* **2020**, *7*, 145. [[CrossRef](#)]
4. Tedjopurnomo, D.A.; Bao, Z.; Zheng, B.; Choudhury, F.M.; Qin, A.K. A Survey on Modern Deep Neural Network for Traffic Prediction: Trends, Methods and Challenges (Extended Abstract). In Proceedings of the 2023 IEEE 39th International Conference on Data Engineering (ICDE), Anaheim, CA, USA, 3–7 April 2023; pp. 3795–3796. [[CrossRef](#)]
5. Li, Y.; Yu, R.; Shahabi, C.; Liu, Y. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv* **2017**, arXiv:1707.01926.
6. Yu, B.; Yin, H.; Zhu, Z. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv* **2017**, arXiv:1709.04875.
7. Zhao, L.; Song, Y.; Zhang, C.; Liu, Y.; Wang, P.; Lin, T.; Deng, M.; Li, H. T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 3848–3858. [[CrossRef](#)]
8. Cai, L.; Janowicz, K.; Mai, G.; Yan, B.; Zhu, R. Traffic transformer: Capturing the continuity and periodicity of time series for traffic forecasting. *Trans. GIS* **2020**, *24*, 736–755. [[CrossRef](#)]
9. Bai, L.; Yao, L.; Li, C.; Wang, X.; Wang, C. Adaptive graph convolutional recurrent network for traffic forecasting. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 17804–17815.
10. Wu, Z.; Pan, S.; Long, G.; Jiang, J.; Zhang, C. Graph wavenet for deep spatial-temporal graph modeling. *arXiv* **2019**, arXiv:1906.00121.
11. Zhang, J.; Zheng, Y.; Qi, D. Deep spatio-temporal residual networks for citywide crowd flows prediction. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31.
12. Chen, W.; Chen, L.; Xie, Y.; Cao, W.; Gao, Y.; Feng, X. Multi-Range Attentive Bicomponent Graph Convolutional Network for Traffic Forecasting. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 3529–3536. [[CrossRef](#)]
13. Geng, X.; Li, Y.; Wang, L.; Zhang, L.; Yang, Q.; Ye, J.; Liu, Y. Spatiotemporal Multi-Graph Convolution Network for Ride-Hailing Demand Forecasting. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 3656–3663. [[CrossRef](#)]
14. Qin, Y.; Fang, Y.; Luo, H.; Zhao, F.; Wang, C. DMGCRN: Dynamic Multi-Graph Convolution Recurrent Network for Traffic Forecasting. *arXiv* **2021**, arXiv:2112.02264.

15. Li, M.; Zhu, Z. Spatial-Temporal Fusion Graph Neural Networks for Traffic Flow Forecasting. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 4189–4196. [[CrossRef](#)]
16. Song, C.; Lin, Y.; Guo, S.; Wan, H. Spatial-Temporal Synchronous Graph Convolutional Networks: A New Framework for Spatial-Temporal Network Data Forecasting. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 914–921. [[CrossRef](#)]
17. He, S.; Luo, Q.; Du, R.; Zhao, L.; He, G.; Fu, H.; Li, H. STGC-GNNs: A GNN-based traffic prediction framework with a spatial-temporal Granger causality graph. *Phys. Stat. Mech. Its Appl.* **2023**, *623*, 128913. [[CrossRef](#)]
18. Ta, X.; Liu, Z.; Hu, X.; Yu, L.; Sun, L.; Du, B. Adaptive Spatio-temporal Graph Neural Network for traffic forecasting. *Knowl.-Based Syst.* **2022**, *242*, 108199. [[CrossRef](#)]
19. Ji, J.; Wang, J.; Huang, C.; Wu, J.; Xu, B.; Wu, Z.; Zhang, J.; Zheng, Y. Spatio-Temporal Self-Supervised Learning for Traffic Flow Prediction. *arXiv* **2022**, arXiv:2212.04475.
20. Liu, X.; Liang, Y.; Huang, C.; Zheng, Y.; Hooi, B.; Zimmermann, R. When do contrastive learning signals help spatio-temporal graph forecasting? In Proceedings of the 30th International Conference on Advances in Geographic Information Systems, Seattle, WA, USA, 1–4 November 2022; pp. 1–12.
21. Shao, Z.; Zhang, Z.; Wang, F.; Xu, Y. Pre-Training Enhanced Spatial-Temporal Graph Neural Network for Multivariate Time Series Forecasting. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '22, Washington, DC, USA, 14–18 August 2022; Association for Computing Machinery: New York, NY, USA, 2022; KDD '22, pp. 1567–1577. [[CrossRef](#)]
22. Hwang, D.; Park, J.; Kwon, S.; Kim, K.M.; Ha, J.W.; Kim, H.J. Self-Supervised Auxiliary Learning with Meta-Paths for Heterogeneous Graphs. In Proceedings of the 34th International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 6–12 December 2020; Curran Associates Inc.: Red Hook, NY, USA, 2020; NIPS'20.
23. 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](#)]
24. Zhao, Z.; Chen, W.; Wu, X.; Chen, P.C.; Liu, J. LSTM network: A deep learning approach for short-term traffic forecast. *IET Intell. Transp. Syst.* **2017**, *11*, 68–75. [[CrossRef](#)]
25. Fu, R.; Zhang, Z.; Li, L. Using LSTM and GRU neural network methods for traffic flow prediction. In Proceedings of the 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), Wuhan, China, 11–13 November 2016; pp. 324–328.
26. Liu, Y.; Zheng, H.; Feng, X.; Chen, Z. Short-term traffic flow prediction with Conv-LSTM. In Proceedings of the 2017 9th International Conference on Wireless Communications and Signal Processing (WCSP), Nanjing, China, 11–13 October 2017; pp. 1–6.
27. Ma, X.; Dai, Z.; He, Z.; Ma, J.; Wang, Y.; Wang, Y. Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction. *Sensors* **2017**, *17*, 818. [[CrossRef](#)]
28. Zonoozi, A.; Kim, J.J.; Li, X.L.; Cong, G. Periodic-CRN: A convolutional recurrent model for crowd density prediction with recurring periodic patterns. In Proceedings of the IJCAI, Stockholm, Sweden, 13–19 July 2018; Volume 18, pp. 3732–3738.
29. Jia, T.; Yan, P. Predicting citywide road traffic flow using deep spatiotemporal neural networks. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 3101–3111. [[CrossRef](#)]
30. Shi, X.; Chen, Z.; Wang, H.; Yeung, D.Y.; Wong, W.K.; Woo, W.c. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 802–810.
31. Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Philip, S.Y. A comprehensive survey on graph neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 4–24. [[CrossRef](#)] [[PubMed](#)]
32. Bruna, J.; Zaremba, W.; Szlam, A.; LeCun, Y. Spectral networks and locally connected networks on graphs. *arXiv* **2013**, arXiv:1312.6203.
33. Atwood, J.; Towsley, D. Diffusion-convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2016**, *29*.
34. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *arXiv* **2017**, arXiv:1710.10903.
35. Tian, C.; Chan, W.K.V. Spatial-temporal attention wavenet: A deep learning framework for traffic prediction considering spatial-temporal dependencies. *IET Intell. Transp. Syst.* **2021**, *15*, 549–561. [[CrossRef](#)]
36. Zhou, Q.; Chen, N.; Lin, S. FASTNN: A Deep Learning Approach for Traffic Flow Prediction Considering Spatiotemporal Features. *Sensors* **2022**, *22*, 6921. [[CrossRef](#)]
37. Jin, G.; Liang, Y.; Fang, Y.; Huang, J.; Zhang, J.; Zheng, Y. Spatio-temporal graph neural networks for predictive learning in urban computing: A survey. *arXiv* **2023**, arXiv:2303.14483.
38. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv* **2014**, arXiv:1412.3555.
39. Xu, M.; Dai, W.; Liu, C.; Gao, X.; Lin, W.; Qi, G.J.; Xiong, H. Spatial-temporal transformer networks for traffic flow forecasting. *arXiv* **2020**, arXiv:2001.02908.
40. Hashemi, S.M.; Hashemi, S.A.; Botez, R.M.; Ghazi, G. Aircraft Trajectory Prediction Enhanced through Resilient Generative Adversarial Networks Secured by Blockchain: Application to UAS-S4 Ehécatl. *Appl. Sci.* **2023**, *13*, 9503. [[CrossRef](#)]
41. Hashemi, S.M.; Hashemi, S.A.; Botez, R.M.; Ghazi, G. A Novel Fault-Tolerant Air Traffic Management Methodology Using Autoencoder and P2P Blockchain Consensus Protocol. *Aerospace* **2023**, *10*, 357. [[CrossRef](#)]
42. Khaled, A.; Elsir, A.M.T.; Shen, Y. TFGAN: Traffic forecasting using generative adversarial network with multi-graph convolutional network. *Knowl.-Based Syst.* **2022**, *249*, 108990. [[CrossRef](#)]

43. Xu, B.; Wang, X.; Liu, Z.; Kang, L. A GAN Combined with Graph Contrastive Learning for Traffic Forecasting. In Proceedings of the 2023 4th International Conference on Computing, Networks and Internet of Things, CNIOT '23, Xiamen China, 26–28 May 2023; Association for Computing Machinery: New York, NY, USA, 2023; CNIOT '23, pp. 866–873. [[CrossRef](#)]
44. Liu, X.; Zhang, F.; Hou, Z.; Mian, L.; Wang, Z.; Zhang, J.; Tang, J. Self-supervised learning: Generative or contrastive. *IEEE Trans. Knowl. Data Eng.* **2021**, *35*, 857–876. [[CrossRef](#)]
45. Banville, H.; Chehab, O.; Hyvärinen, A.; Engemann, D.A.; Gramfort, A. Uncovering the structure of clinical EEG signals with self-supervised learning. *J. Neural Eng.* **2021**, *18*, 046020. [[CrossRef](#)] [[PubMed](#)]
46. Chung, Y.A.; Hsu, W.N.; Tang, H.; Glass, J. An unsupervised autoregressive model for speech representation learning. *arXiv* **2019**, arXiv:1904.03240.
47. Bai, J.; Wang, W.; Zhou, Y.; Xiong, C. Representation learning for sequence data with deep autoencoding predictive components. *arXiv* **2020**, arXiv:2010.03135.
48. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.