frontiers in Public Health

ORIGINAL RESEARCH
published: 25 January 2022
doi: 10.3389/fpubh.2021.804298

A Deep Learning Framework About Traffic Flow Forecasting for Urban Traffic Emission Monitoring System

Baozhen Yao[1], *Ankun Ma*[1], *Rui Feng*[1], *Xiaopeng Shen*[2], *Mingheng Zhang*[1]* and *Yansheng Yao*[3]

[1] State Key Laboratory of Structural Analysis for Industrial Equipment, School of Automotive Engineering, Dalian University of Technology, Dalian, China, [2] CIECC Overseas Consulting Co., Ltd., Beijing, China, [3] School of Mechanical and Electrical Engineering, Anhui Jianzhu University, Hefei, China

As urban traffic pollution continues to increase, there is an urgent need to build traffic emission monitoring and forecasting system for the urban traffic construction. The traffic emission monitoring and forecasting system's core is the prediction of traffic emission's evolution. And the traffic flow prediction on the urban road network contributes greatly to the prediction of traffic emission's evolution. Due to the complex non-Euclidean topological structure of traffic networks and dynamic heterogeneous spatial-temporal correlations of traffic conditions, it is difficult to obtain satisfactory prediction results with less computation cost. To figure these issues out, a novel deep learning traffic flow forecasting framework is proposed in this paper, termed as Ensemble Attention based Graph Time Convolutional Networks (EAGTCN). More specifically, each component of our model contains two major blocks: (1) the global spatial patterns are captured by the spatial blocks which are fused by the Graph Convolution Network (GCN) and spatial ensemble attention layer; (2) the temporal patterns are captured by the temporal blocks which are composed by the Time Convolution Net (TCN) and temporal ensemble attention layers. Experiments on two real-world datasets demonstrate that our model obtains more accurate prediction results than the state-of-the-art baselines at less computation expense especially in the long-term prediction situation.

Keywords: urban traffic construction, traffic flow analysis, deep learning, graph, prediction model

OPEN ACCESS

Edited by:
Yibin Ao,
Chengdu University of Technology, China

Reviewed by:
Weitiao Wu,
South China University of Technology, China
Fang Zong,
Jilin University, China

****Correspondence:***
Mingheng Zhang
zhangmh@dlut.edu.cn

Specialty section:
This article was submitted to Environmental health and Exposome, a section of the journal Frontiers in Public Health

Received: *29 October 2021*
Accepted: *10 December 2021*
Published: *25 January 2022*

Citation:
Yao B, Ma A, Feng R, Shen X, Zhang M and Yao Y (2022) A Deep Learning Framework About Traffic Flow Forecasting for Urban Traffic Emission Monitoring System. Front. Public Health 9:804298. doi: 10.3389/fpubh.2021.804298

INTRODUCTION

With the rapid development of urban traffic construction, traffic emission has attracted more and more attention from public. The traffic emission contains carbon monoxide, nitrogen oxides and particulate matter, which are the main causes of smog and photochemical smog pollution (1, 2). Public's health pays much price for the traffic emission (2–5).

As a result, there is a need for urban traffic construction to establish an effective environmental monitoring and early warning system (2), whose core is the accurate prediction of the evolution of traffic emission. The trend of traffic emission evolution is mainly affected by traffic conditions including traffic flow, traffic velocity and road occupancy (6). Predicting the emission of traffic road networks means predicting trends of those traffic condition variables (7). Therefore, the accurate and efficient predictions of traffic condition variables' trends can provide scientific foundation for predicting the evolution of urban traffic emission. The task of traffic condition variables' predictions is to provide punctual, continuous and precise traffic condition variables prediction information based on the past measurements of traffic and the underlying road networks. Among those traffic condition variables, traffic flow is harder to predict (8). The challenges of traffic flow prediction can be summarized into the following two parts: accuracy and efficiency (9).

First, the accuracy problem origins from non-Euclidean topological structure of traffic networks, the stochastic characteristics of the non-stationary traffic patterns and inherent difficulties in multiple steps ahead prediction (10). The key to improve the accuracy of prediction is to capture simultaneously the dynamic heterogeneous spatial-temporal correlations of traffic conditions. On the one hand, the spatial correlations can not only be found at a local scale but also in a wide range of the traffic networks. Two distant roads in a traffic network might have high correlations, too (11). And different locations have different impacts on the study location (12). Moreover, since traffic is constantly evolving, the spatial correlations are dynamic (10). For instance, in the morning, the correlations between a residential area and a business center could be strong; whereas in late evening, the correlations between those might be very weak (13). On the other hand, traffic observations exhibit autocorrelations at the adjacent time intervals and show cyclical patterns. Traffic observation variations are affected by vehicular traffic lights, changes in weather, and other factors (9). Some of these factors play a long-term decisive role, making the variations show specific trends and certain regularities, while others play a short-term role, introducing some uncertainty into the variations. And the temporal correlations and periodicity vary in different time-of-day (10), which brings great difficulties to the prediction of traffic flow. Secondly, The efficiency problems of predictions origin from the scale of traffic flow predictions. On the spatial parts, the large size of traffic networks requires more sensors to detect traffic status. On the temporal parts, the application of traffic flow predictions needs longer forecasting steps. Both two parts increase the scale of predictions, which will cost more time.

Historically, related field researchers have exploited statistical methods, machine learning and deep learning approaches for modeling the complex temporal-spatial patterns of traffic flow forecasting problems. The widely used statistical methods include autoregressive integrated moving average (ARIMA) (14–16), Kalman filtering (17), Markov chain (18), and exponential smoothing methods (19–21). Those classical statistical models were purely inductive methods, which placed strong stationary assumptions on the traffic flow sequence. However, it was difficult to satisfy these assumptions in the real world due to the inherent complexity of traffic data. Therefore, those classical statistical methods did not have enough capability to capture dynamic patterns of traffic flow. And each step prediction was based on the prior predictions, which led to the propagation and accumulation of errors.

Along with the development of the computing device and information explosion, machine learning models have caused wide public concerns, including k-nearest neighbors (22), support vector regression (23) and random forest (24). Complex nonlinear traffic data can be regressed by those ML methods, but the premises were to conduct detailed feature engineering, which was critical but difficult. Furthermore, the power of capturing the complex non-stationary temporal patterns was limited by ML models' shallow architectures, especially for long-term forecasting (10).

Deep Learning (DL) network is an effective tool for regression problems like the traffic flow forecasting. This method aims to automatically identify patterns and extract features from the historical information by constructing an appropriate parameter space. The DL models have made breakthroughs in many domains, such as speech recognition and image processing. Those progress made by DL has drawn substantial interests among transportation researchers and they have been trying to apply deep learning models in many traffic prediction problems. Initially, the traffic status data was simply treated as normal temporal sequence and was predicted by classical Recurrent Neural Networks (RNNs) like Long Short Term Memory (LSTM) and Gated Recurrent Unit neural networks (GRU) (25, 26). Those networks neglected the modeling of traffic data's spatial attributes. Subsequently, some related researchers began to take both the spatial patterns and temporal patterns into consideration. Convolutional Neural Networks (CNN) which took charge of extracting spatial dependencies from traffic data was introduced into RNNs. Du et al. proposed a hybrid deep learning framework which consists of CNN and RNNs for short-term traffic flow forecasting (27). The Fusion Convolutional Long Short Term Memory Network (FCL-Net) (28) proposed by Ke et al. stacked and fused multiple LSTM layers, standard LSTM layers and CNN layers to capture the spatial-temporal characteristics of explanatory variables. Shi et al. constructed the convolutional LSTM model by combining the normal fully-connected LSTM and convolutional layers (29). Those models just treated the traffic status data at a certain time slice as an image. CNN as a typical deep neural network can effectively capture the spatial features of grid data. However, due to the non-Euclidean topology of the traffic networks, CNN actually does not have enough ability to extract the spatial patterns of the road networks. Recently, the Graph Convolution Network (GCN) has been widely used because this network can generalize the traditional convolution operations to non-Euclidean graph structure data (30). Many forecasting models based on GCN were proposed. Li et al. proposed graph and attention-based long short-term memory network (GLA) which was composed of GCN and LSTM (31). Zhu et al. proposed a new traffic flow prediction method based on RNN-GCN and the Belief Rule Base (BRB) (32). Spatial Temporal Graph Convolutional Networks (STGCN) was proposed by Yu et al., which combined GCN and 1D-CNN to capture spatial-temporal patterns (5). Attention Based Spatial Temporal Graph Convolutional Networks (ASTGCN) (12) presented by Guo et al. confused STGCN with attention layers which were used to capture the dynamic spatial correlations among nodes relying only on traffic flow data. However, these GCN based methods still have many problems. On the one hand, those methods could not comprehensively capture spatial-temporal dynamic heterogeneous features of traffic data which have significant influence on the traffic forecasting issues. On the other hand, the parts of those models which were responsible

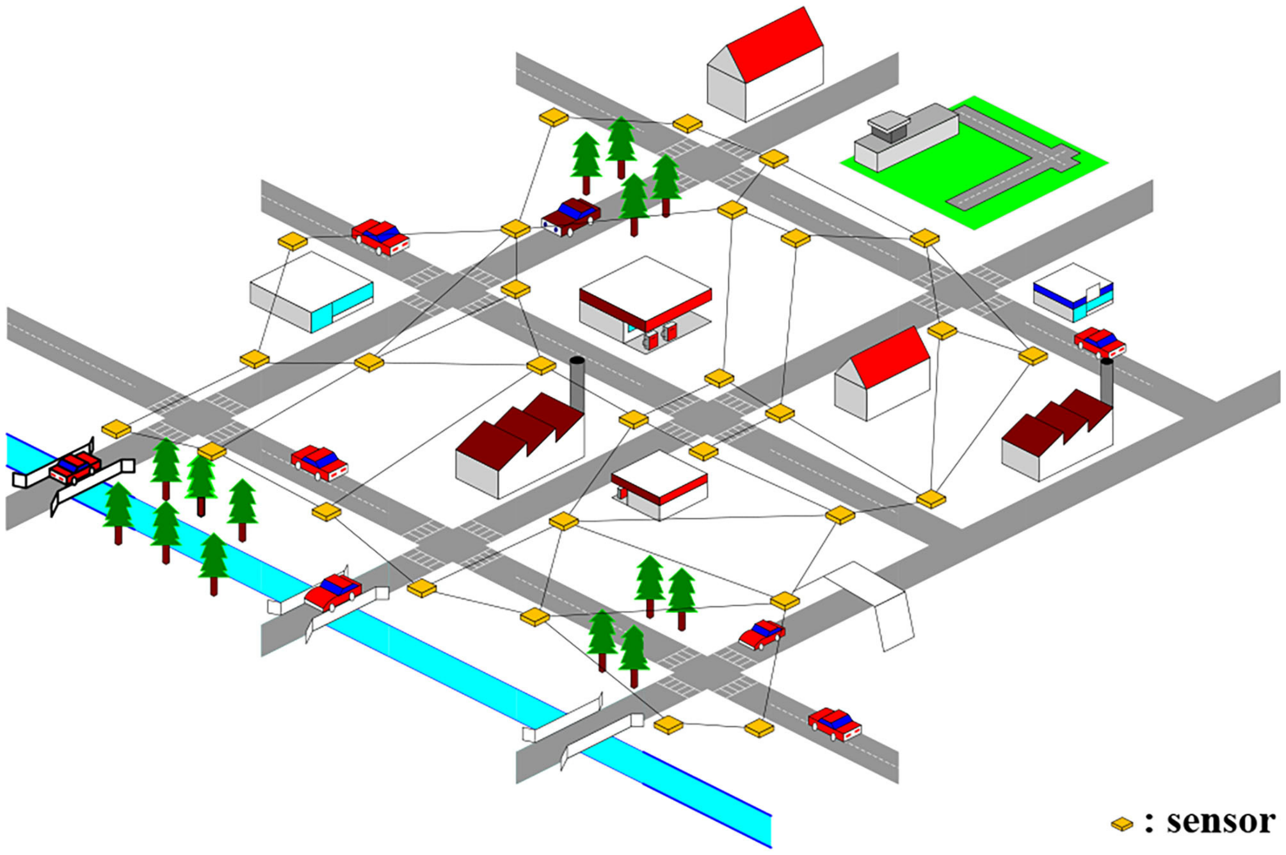


FIGURE 1 | An example of topology graph of traffic network.

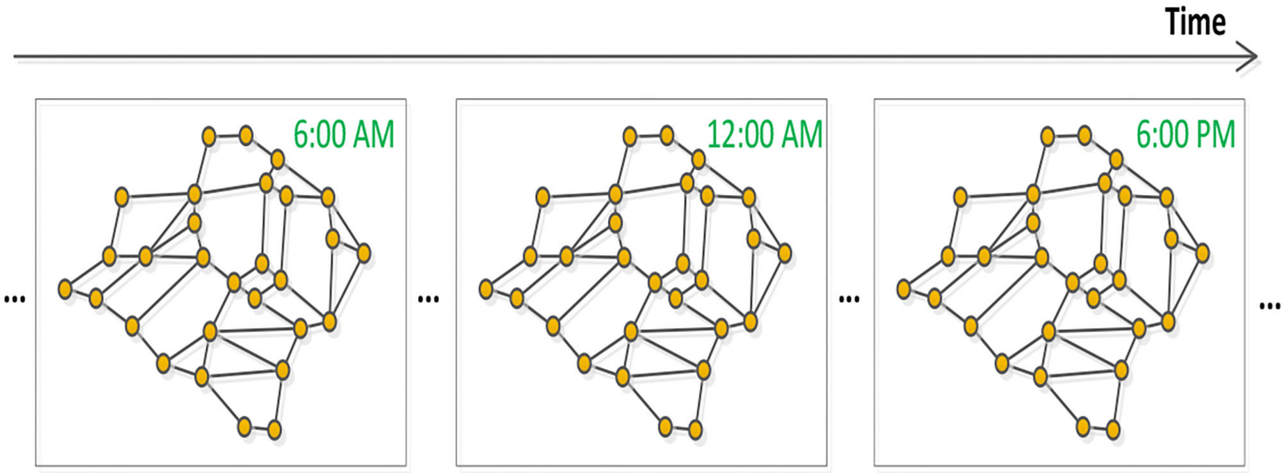


FIGURE 2 | Traffic temporal-spatial graph sequence.

to capture temporal patterns were constructed by either RNNs or normal 1D CNN. The models constructed by RNNs did well in long-term dependencies' capturing but required too much computing time and suffered from gradient exploding or vanishing (33). The models constructed by 1D CNN were able to decrease the computation expense but needed more stacked layers to capture long-term temporal dependencies. DL models with too deep convolution layers might lose some key

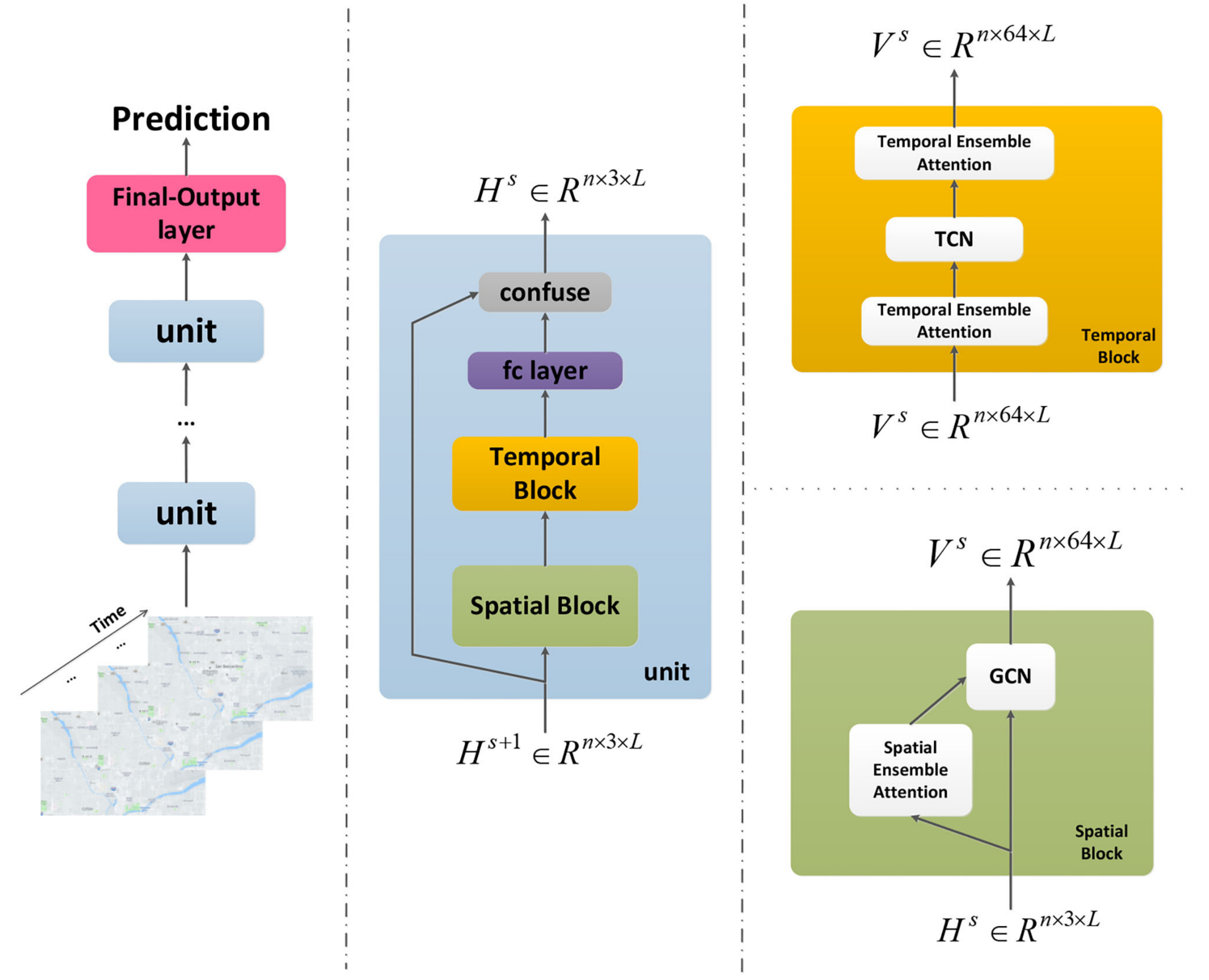

Prediction
Final-Output layer
unit
...
unit
Time
...
...
$H^{s} \in R^{n\times3\times L}$
confuse
fc layer
Temporal Block
Spatial Block
unit
$H^{s+1} \in R^{n\times3\times L}$
$V^{s} \in R^{n\times64\times L}$
Temporal Ensemble Attention
TCN
Temporal Ensemble Attention
Temporal Block
$V^{s} \in R^{n\times64\times L}$
$V^{s} \in R^{n\times64\times L}$
GCN
Spatial Ensemble Attention
Spatial Block
$H^{s} \in R^{n\times3\times L}$

FIGURE 3 | The framework of our model.

information in long-term forecasting, which results in the decline of forecasting accuracy.

In the paper, a novel deep learning model named Ensemble Attention Graph Time Convolutional Networks (EAGTCN) is proposed to predict traffic flow in the road network dimension. This model can capture more comprehensive dynamic heterogeneous spatial-temporal features of traffic data effectively and efficiently. Significantly, the model is applicable to the other traffic condition variables' forecasting and provides the solid foundation for traffic emission's prediction and monitoring. The main contributions of this paper are summarized as follows:

(1) We propose an ensemble attention mechanism which is able to dig out the global dynamic heterogeneous spatial-temporal correlations from traffic sequence.
(2) TCN is applied to capture basic temporal dependencies. The TCN has much longer effective memory while can be trained fast.
(3) Our model is evaluated with the real-world traffic data, showing that our model outperforms than the state-of-art, especially in the long-term prediction situation.

PROBLEM DESCRIPTION

Traffic Network Based on Graph

Normally, a spatial-temporal graph is composed of nodes and edges which connect those nodes. Spatial-temporal graph is defined as $G^{(\tau)} = \left(V^{(\tau)}, E^{(\tau)}, V^{(\tau)}_{attr}, E^{(\tau)}_{attr}\right)$, where τ denotes the time slice τ. $V^{(\tau)} = \{V_1, V_2, V_3, \cdots, V_j\}$ represents all the nodes of the graph at time slice τ. $E^{(\tau)} = \{(e_1, r_1, s_1), (e_2, r_2, s_2), \cdots, (e_k, r_k, s_k)\}$ denotes the

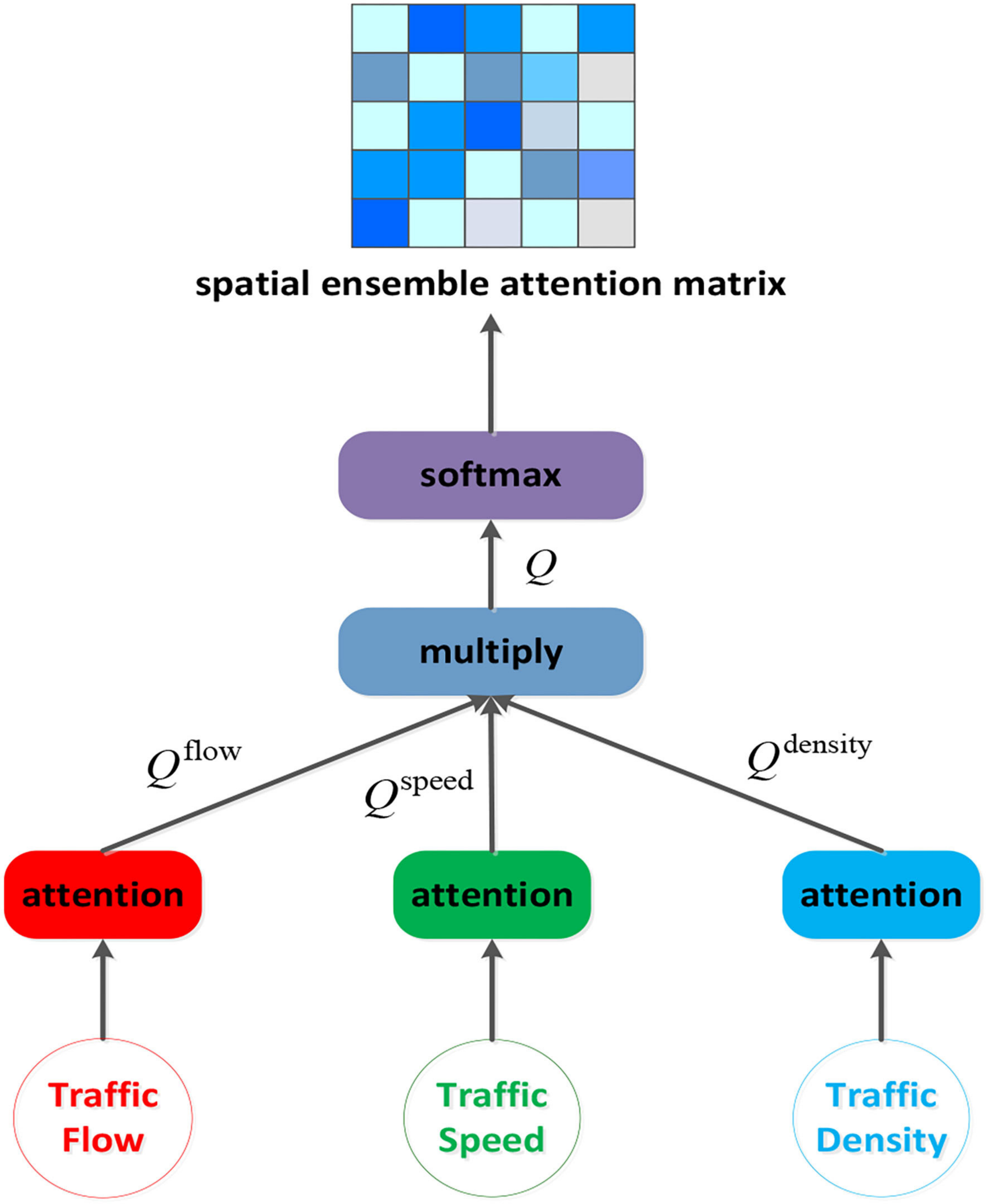


FIGURE 4 | The structure of spatial ensemble attention layer.

whole edges of the graph at time slice τ. $e_k = v_{ij}$ represents the edge k between node i and node j. r_k, s_k denote the edge k's receiving node and sending node respectively. $V_{attr}^{(\tau)} \in R^{n \times c}$ represents the features of all nodes at time slice τ, where $n = \left|V^{(\tau)}\right|$ denotes the number of nodes and c denotes the dimension of a node feature vector. $E_{attr}^{(\tau)} \in R^{u \times d}$ represents the features of all edges at time sliceτ, where $u = \left|E^{(\tau)}\right|$ denotes the number of edges, and d denotes the dimension of a edge feature vector. The structure of graph $G^{(\tau)}$ can be represented by $\left(V^{(\tau)}, E^{(\tau)}\right)$. The adjacency matrix $A = \left(A_{ij}\right) \in R^{n \times n}$ can be transferred from $\left(V^{(\tau)}, E^{(\tau)}\right)$, in which $A_{ij} = 1$ if there is an edge between node i and node j and $A_{ij} = 0$ otherwise ($A_{ii} = 0$).

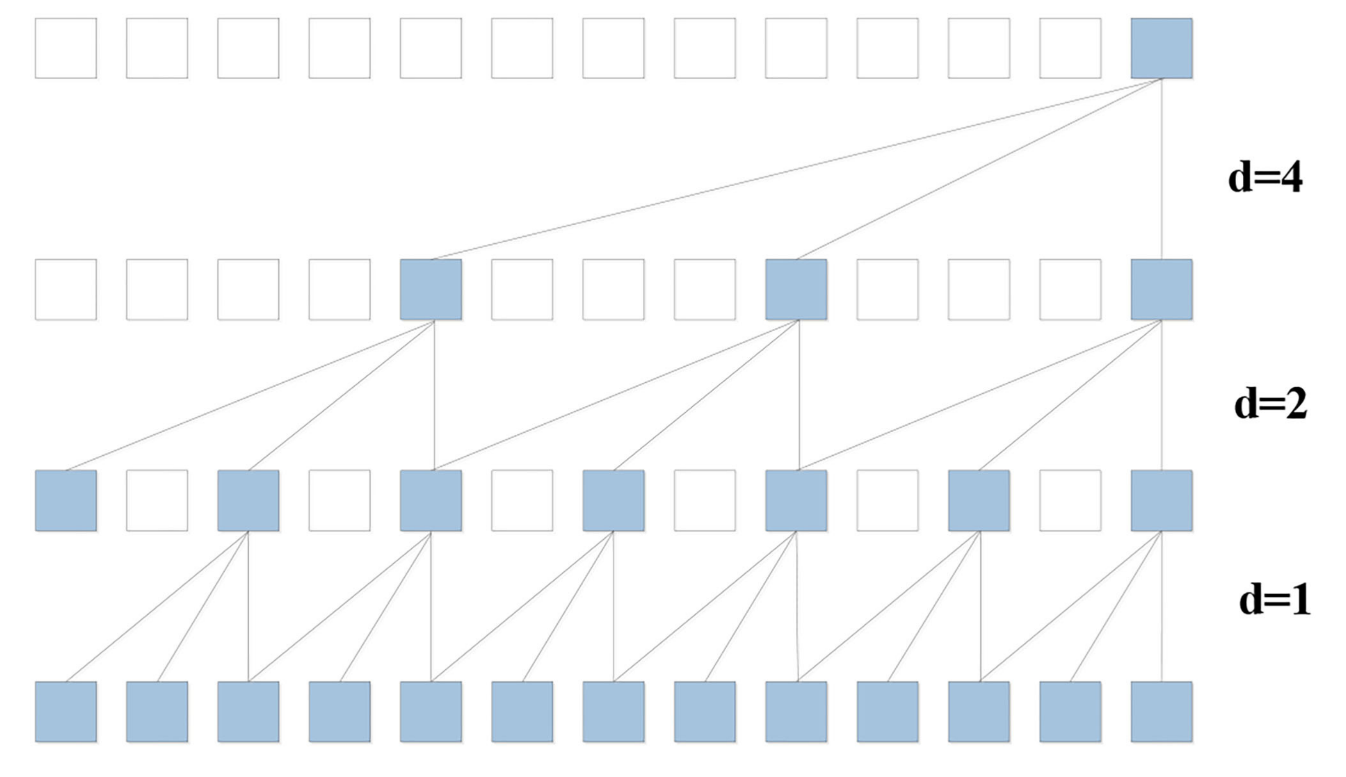


FIGURE 5 | Time convolution net with kernel size three and dilation factor $d = 1, 2, 4$.

As is shown by (**Figure 1**), the sensors deployed on the traffic network which collected traffic data at fixed time intervals form a non-Euclidean topological graph naturally (34). We use nodes to represent the locations of traffic sensors, and the road segments connecting traffic sensors are treated as edges in graph. After that, the traffic network can be abstracted into a topology graph. Due to the traffic network structure's stability (35), we can keep the structure of its topology graph fixed, which means that $V^{(\tau)}$ and $E^{(\tau)}$ do not change over time. We define the traffic network at time slice τ as an undirected spatial-temporal graph $G^{(\tau)} = \left(X^{(\tau)}, V, E\right)$. Then the traffic spatial-temporal sequence G can be defined as $\left\{X^{(1)}, X^{(2)}, X^{(3)}, \cdots, X^{(\tau)}, \cdots, X^{(t)}; V, E\right\}$, where $X^{(\tau)} \in R^{n\times c}$ denotes the values of all the features of traffic sensors at time slice τ.

Traffic Flow Forecasting Problem

Based on the above analysis, the traffic flow forecasting problem can be defined as a temporal-spatial sequence prediction problem based on graphs, which is shown by (**Figure 2**).

Given the previous t time slices traffic status $X = \left(X^{(1)}, X^{(2)}, X^{(3)}, \cdots, X^{(\tau)}, \cdots, X^{(t)}\right) \in R^{n\times c\times t}$ and the graph structure V, E, our task is to predict the future q time slices which can be denoted as:

$$\left\{X^{(1)}, X^{(2)}, \cdots, X^{(\tau)}, \cdots, X^{(t)}; V, E\right\} \overset{MAP}{\rightarrow} \left(X^{(t+1)}, X^{(t+2)}, \cdots, X^{(t+q)}\right) \quad (1)$$

METHODOLOGY

In this section, we elaborate the framework of our model and its basic modules (the spatial patterns modeling part and the temporal patterns modeling part).

Network Architecture

The model presented here is an end-to-end framework which is showed by (**Figure 3**). It has N units and a final-output layer. The final-output layer can generate the final prediction results by integrating comprehensive features. Each unit consists of a spatial block, a temporal block, a fully connected layer and a confuse layer. There is a GCN module and a spatial ensemble attention module in the spatial block. Each temporal block contains two temporal ensemble attention modules and a TCN module in the middle. In the confuse layer, residual connection is applied to optimize the training efficiency and reshape the output of this unit. The dynamic heterogeneous spatial-temporal patterns of the traffic flow are going to be captured elaborately by the overall framework. The main parts of this architecture will be described in details as following sections.

Spatial Patterns Modeling

Graph Convolution

Traffic prediction is a typical task where data are generated from non-Euclidean domains, and the traffic network can be represented as graphs naturally where nodes have complex spatial correlations. Those frequently-used deep learning methods such

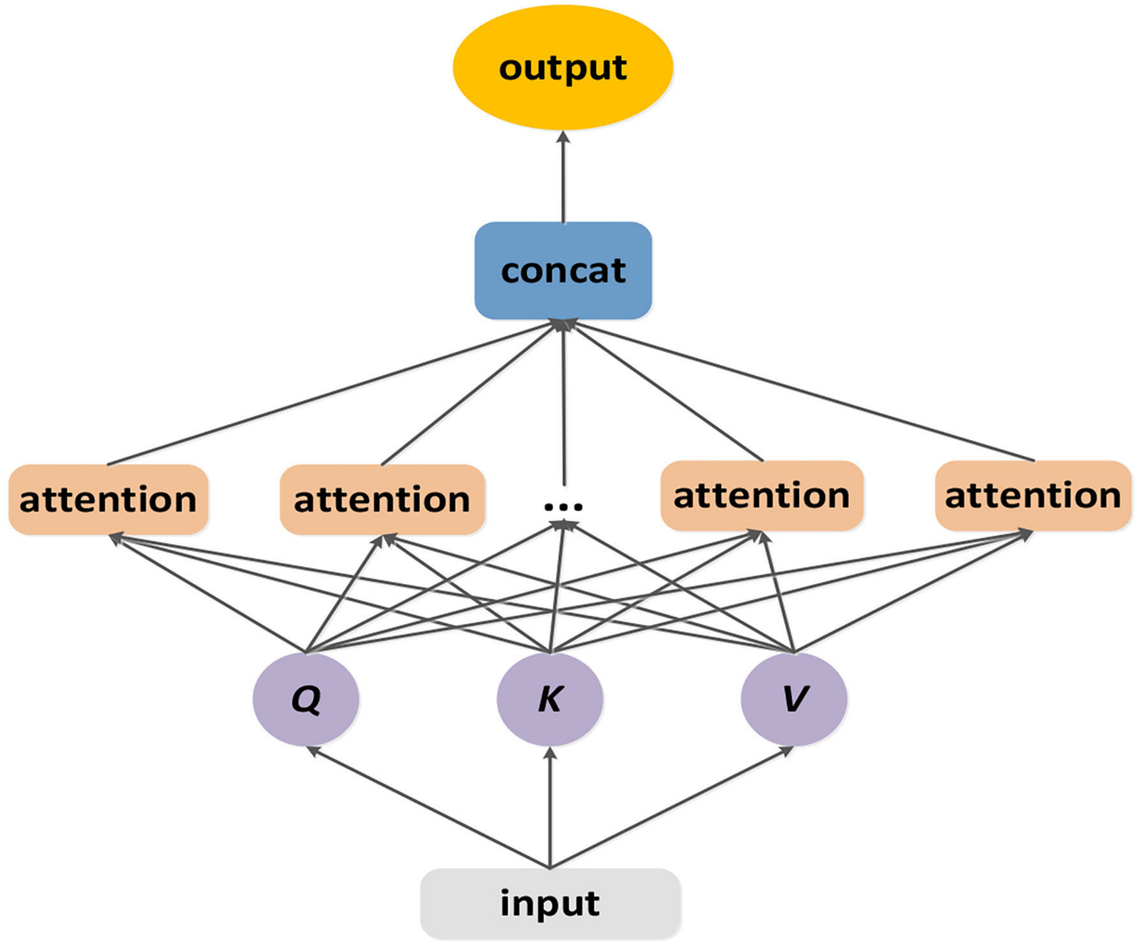


FIGURE 6 | The structure of temporal ensemble attention layer.

TABLE 1 | Traffic flow forecasting comparison in the next 15, 30 and 60 min on PeMSD08.

Models	MAE			RMSE		
	15 min	30 min	60 min	15 min	30 min	60 min
SVR	18.98	19.34	23.96	30.25	27.80	34.17
RF	19.52	20.02	24.26	30.34	28.25	33.97
GRU	21.51	23.70	26.14	31.66	34.51	37.39
LSTM	20.20	20.75	23.14	29.40	30.28	34.28
Seq2Seq	22.12	22.81	24.72	34.96	35.93	38.38
ASTGCN	17.32	18.65	21.77	25.30	27.52	31.68
Ours	16.59	17.58	18.96	24.28	26.89	28.72
Improvement	2.14%	5.74%	12.91%	4.03%	5.19%	9.34%

as CNNs which are born to deal with Euclidean data cannot solve graph-based problems nicely. Hence, in order to model basic spatial dependencies between nodes per time slice of the sequence better, we apply the GCN method.

The GCN method applied here defines convolution filters from the view of signal processing (36) where the convolution operation is treated as removing noises from graph signals. We use the normalized graph Laplacian matrix to represent the graph (36, 37), defined as $L = I_n - D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$, where I_n is a n-dimension identity matrix, $A \in R^{n \times n}$ is the adjacent matrix and D is a diagonal matrix of node degrees ($D_{ii} = \sum_j A_{ij}$). Due to normalized graph Laplacian matrix's real symmetric positive semidefinite property, it could be decomposed as$L = U\Lambda U^T$, where eigenvectors matrix

$U = [u_0, u_1, u_2, \cdots, u_{n-1}] \in R^{n\times n}$ forms an orthonormal space and $\Lambda$is the matrix of eigenvalues ($\Lambda_{ii} = \lambda_i$). On the other hand, $X_{:,:,\tau:\tau+1} = \left[x^{(0)}, x^{(1)}, x^{(2)}, \cdots, x^{(i)}, \cdots, x^{(N-1)}\right] \in R^{n\times 3\times 1}$ represents all the signals of the graph in time slice τ, where $x^{(i)}$ is the signal value of i_{th} node. Based on U from the decomposition of L, we can define the graph Fourier transformation to $X_{:,:,\tau:\tau+1}$ as $\widehat{X_{:,:,\tau:\tau+1}} = F\left(X_{:,:,\tau:\tau+1}\right) = U^T X_{:,:,\tau:\tau+1}$ and the inverse graph Fourier transformation can be defined as $F^{-1}\left(\widehat{X_{:,:,\tau:\tau+1}}\right) = U\widehat{X_{:,:,\tau:\tau+1}}$, where $\widehat{X_{:,:,\tau:\tau+1}}$ is the signal after graph Fourier transformation. The graph Fourier transformation is able to reflect the graph signal $X_{:,:,\tau:\tau+1}$ to an orthonormal space built by eigenvectors matrix U.

Based on this, we can define the graph convolution operation as:

$$\begin{aligned} X_{:,:,\tau:\tau+1} *_G g &= F^{-1}\left(F\left(X_{:,:,\tau:\tau+1}\right) \odot F\left(g\right)\right) \\ &= U\left(U^T X_{:,:,\tau:\tau+1} \odot U^T g\right) \end{aligned} \quad (2)$$

where $\underset{G}{*}$ is the graph convolution operation, and $\odot$ denotes the element-wise product. We usually treat $\operatorname{diag}\left(U^T g\right)$ as the graph convolution filter g_θ which is made up of some learnable parameters. Then the formula (1) could be simplified as below:

$$X_{:,:,\tau:\tau+1} *_G g = U g_\theta U^T X_{:,:,\tau:\tau+1} \quad (3)$$

However, the eigen-decomposition of the Laplacian matrix requires $O\left(n^3\right)$ computational complexity, which brings too much computation cost. Therefore, Chebyshev polynomials are applied to reduce the computation expense of the graph convolution operations (38). We use the Chebyshev polynomials of the diagonal matrix of eigenvalues (30) to approximate the filter $g_\theta = \sum_{i=0}^{k} \theta_i T_i\left(\tilde{\Lambda}\right)$, where $\tilde{\Lambda} = \Lambda/\lambda_{max} - I_n$, $\theta_i \in R^k$ denotes the co-efficient and λ_{max} is the maximum eigenvalue of the Laplacian matrix. The Chebyshev polynomials are defined recursively by $T_i(\varphi) = 2\varphi T_{i-1}(\varphi) - T_{i-2}(\varphi)$ with $T_0(\varphi) = 1$ and $T_1(\varphi) = \varphi$. As a result, the graph convolution of signal $X_{:,:,\tau:\tau+1}$ with the defined filter g_θ can finally be defined as:

$$\begin{aligned} X_{:,:,\tau:\tau+1} *_G g &= U\left(\sum_{i=0}^{k} \theta_i T_i\left(\tilde{\Lambda}\right)\right) U^T X_{:,:,\tau:\tau+1} \\ &= \left(\sum_{i=0}^{k} \theta_i T_i\left(\tilde{L}\right)\right) X_{:,:,\tau:\tau+1} \end{aligned} \quad (4)$$

where $\tilde{L} = 2L/\lambda_{max} - I_n$.

Spatial Ensemble Attention

In the traffic networks, the traffic conditions of one node exert significant but different influence on others'. This kind of traffic heterogeneous spatial correlations can be reflected in the traffic speed, the traffic flow and the traffic density. According to the traffic flow theory, there is a complex relation among them (39), and a multitude of models describing relationships between traffic flow and the other two variables have been developed over the years (40–43). We are supposed to ensemble those factors together to get more comprehensive spatial correlations among nodes, rather than relying solely on the traffic flow.

TABLE 2 | Traffic flow forecasting comparison in the next 60 min on PeMSD04 and PeMSD08.

Data	Models	60 min	
		MAE	RMSE
PEMS-08	SVR	23.96	34.17
	Random Forest	24.26	33.97
	GRU	26.14	37.39
	LSTM	23.14	34.28
	Seq2Seq	24.72	38.38
	ASTGCN	21.77	31.68
	Ours	18.96	28.72
PEMS-04	SVR	29.49	41.49
	Random Forest	29.57	41.28
	GRU	28.79	44.02
	LSTM	26.59	40.36
	Seq2Seq	26.49	42.08
	ASTGCN	26.00	41.12
	Ours	25.51	37.15

TABLE 3 | Training efficiency comparison.

Model	Average training time (s/epoch)
ASTGCN	329.84
Our Model	256.07

Considering of that, a spatial ensemble attention mechanism is proposed which aims to dig out richer dynamic heterogeneous relationships between nodes by packing each factor's attention matrix together.

$$Q^{(c)} = V_s^{(c)} \sigma\left(\left(X_{:,c:c+1,:} W_1^{(c)}\right) W_2^{(c)} + b_s^{(c)}\right) \quad (5)$$

$$Q = \prod_{c=0}^{2} Q^{(c)} \quad (6)$$

$$Q_{i,j}{}' = \operatorname{softmax}\left(Q_{i,j}\right) \quad (7)$$

where $X_{:,c:c+1,:} \in R^{n\times 1\times L}$ denotes one feature slice from the spatial ensemble attention module's input $X_{:,:,:} \in R^{n\times 3\times L}$, where n is the number of nodes, three is the number of input graph signal's features (the traffic flow, the traffic speed and the traffic density), and L is the length of the time length. $V_s^{(c)} \in R^{n\times n}$, $b_s^{(c)} \in R^{n\times n}$, $W_1^{(c)} \in R^L$ and $W_2^{(c)} \in R^{L\times n}$ are the learnable parameters. We construct the spatial ensemble attention matrix $Q \in R^{n\times n}$ by multiplying the three factors' attention matrix $Q^{(c)}$. After the ensemble operation, the softmax function is used to make sure the spatial ensemble attention matrix's elements sum

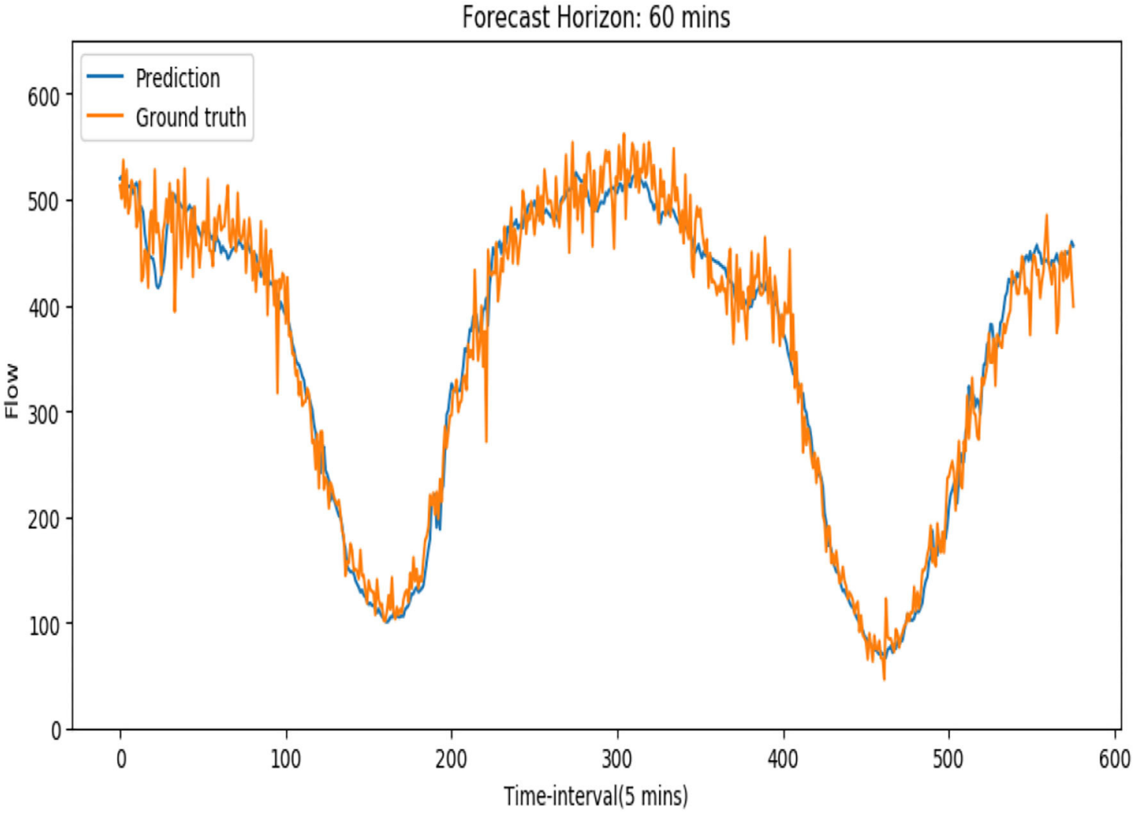


FIGURE 7 | Traffic flow prediction and the corresponding ground truth of a sensor during 2 days.

to one. The spatial ensemble attention operation is presented by (**Figure 4**).

After this, we accompany the Chebyshev polynomials filters with the spatial ensemble attention matrix Q'. Then the $\mathrm{s_{th}}$ spatial block's output can be obtained by:

$$\mathrm{V}^{\mathrm{s}}_{:,:,:} = \sigma(\left(\sum_{m=0}^{k} \theta_{\mathrm{m}} T_{\mathrm{m}}\left(\tilde{L}\right) \odot \mathrm{Q}')H^{s}_{:,0:1,:}\right) \tag{8}$$

where s is the index of the spatial block and the σ function is Rectified Linear Unit (ReLU). $\mathrm{H}^{\mathrm{s}}_{:,0:1,:} \in \mathrm{R}^{\mathrm{n}\times 1\times \mathrm{L}}$ is the traffic flow slice of the $\mathrm{s_{th}}$ spatial block's input $\mathrm{H}^{\mathrm{s}}_{:,:,:} \in \mathrm{R}^{\mathrm{n}\times 3\times \mathrm{L}}$, and $\mathrm{V}^{\mathrm{s}}_{:,:,:}$ denotes the $\mathrm{s_{th}}$ spatial block's output signal.

Temporal Patterns Modeling

Time Convolution

In the temporal trend analysis, the RNN-based methods are applied extensively. However, the recurrent networks are still stuck in some problems, such as exploding/vanishing gradients and time-consuming iterations (44). On the other hand, the traditional 1D convolution method does not have enough ability to memorize long historical information.

Considering the issues above, the TCN frame (45), a simple but highly effective network, is employed to capture the basic temporal dependencies of traffic flow. As is showed in **Figure 5**, this special designed network supports parallel training procedures to improve training efficiency. Meanwhile, this simple but effective network has ability to look far enough into the past to improve prediction accuracy.

The convolution operation in TCN is dilated causal convolution, which is a variant of causal convolution. Suppose input $\mathrm{X}_{\mathrm{v}:(\mathrm{v}+1),:,:}$ is a traffic flow temporal sequence at the node v and there is a filter $\mathrm{f}: \left\{0, 1, 2, \cdots, \mathrm{k}-1\right\}$. The dilated causal convolution operation on the u element of the sequence can be defined as:

$$(\mathrm{X}_{\mathrm{v}:(\mathrm{v}+1),:,:}{}^{*}{}_{\mathrm{d}}\mathrm{f})\,(\mathrm{u}) = \sum_{\mathrm{i}=0}^{\mathrm{k}-1} \mathrm{f}\,(\mathrm{i})\,\mathrm{X}_{\mathrm{v}:(\mathrm{v}+1),:,:}\left(\mathrm{u}-\mathrm{id}\right) \tag{9}$$

where u − id accounts for the direction of the past and d is the dilation factor. The dilation factor d is the key parameter to control the distance between every two adjacent filter taps. When the dilation factor is set to one, the dilation causal convolution reduces to the casual convolution. We increase the dilation factor d exponentially with the depth of the network, and the receptive field of the model grows exponentially. This ensures that the long effective historical information can always be captured by some filters.

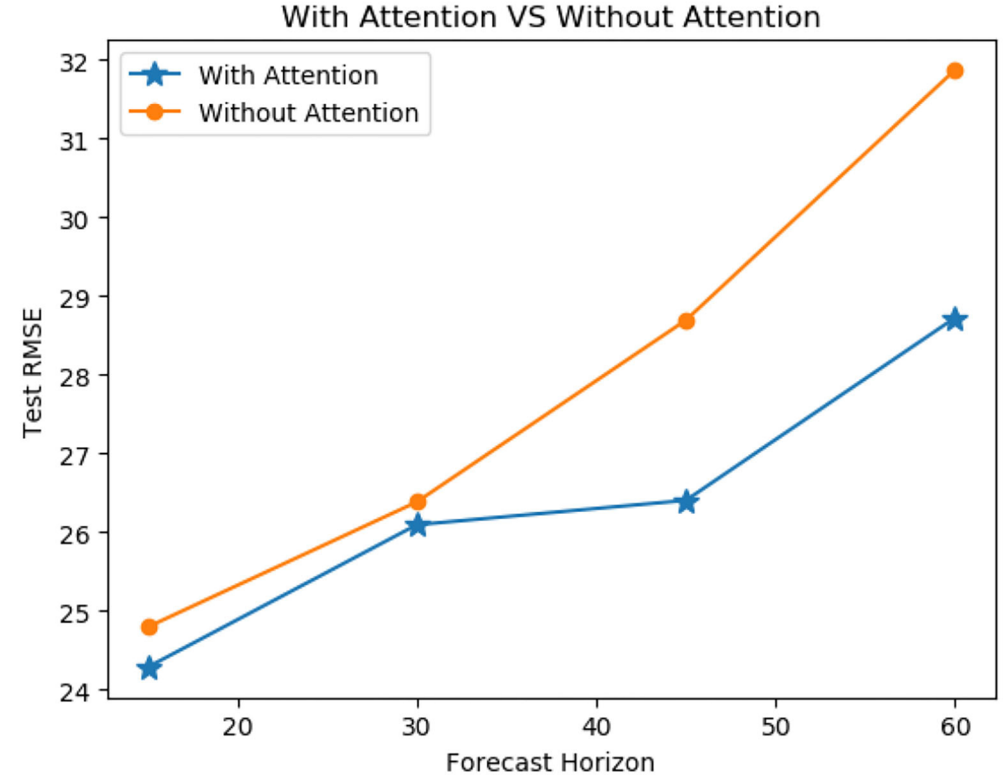


FIGURE 8 | Average performance comparison of models with or without attention mechanism on PeMSD8.

Temporal Ensemble Attention

In the temporal dimension, the previous traffic conditions have significant but different influence on the following conditions, too. And the correlations among different time slices are complex. Motivated by the transformer framework's multi-head attention (46), we designed a temporal ensemble attention mechanism which is showed by (**Figure 6**). The temporal ensemble attention operation can capture more comprehensive heterogeneous temporal correlations by expanding bigger feature space at a less computation cost.

The temporal ensemble attention operation is designed based on the self-attention mechanism.

$$SA = \text{selfattention}\,(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{C}}\right)(V) \quad (10)$$

where the $Q \in R^{n \times L \times c}$ is the query matrix, the $K \in R^{n \times L \times c}$ is the key matrix and the $V \in R^{n \times L \times c}$ is the value matrix. The matrixes Q, K, V correspond with the input of the temporal ensemble attention module. The output SA is computed as a weight sum of the value matrix, where the weight assigned to each element is obtained by computing the compatibility of the query with the corresponding key.

Due to the complex relationships among the time slices, there is a need to expand the feature space to represent the traffic status. But too many features would increase computational expense. Learning from the multi-head attention (46), we linearly project the query, key and value h times. On each of these projected versions of query, key and value, the self-attention operation is applied in parallel. We contact those outputs and project it linearly to get the final outcome.

$$\begin{aligned} Q_i &= QW_i^Q && (11)\\ K_i &= KW_i^K && (12)\\ V_i &= VW_i^V && (13)\\ SA_i &= \text{selfattention}\,(Q_i, K_i, V_i)\\ &= \text{softmax}\left(\frac{Q_i K_i^T}{\sqrt{C_k}}\right)(V_i) && (14)\\ TEA\,(Q, K, V) &= \text{Concat}\,(SA_1, SA_2, SA_3 \cdots, SA_i, \cdots, SA_h)\\ & W^o && (15) \end{aligned}$$

where $W_i^Q \in R^{c \times C_k}$, $W_i^K \in R^{c \times C_k}$, $W_i^V \in R^{c \times C_k}$, $C_K = c/h$, $W^o \in R^{c \times c}$. The temporal ensemble attention can not only confuse information from subspace but also balance the conflicts between model's express ability and the computation cost.

EXPERIMENTS

In this section, in order to evaluate the performance of our model, we verify it on two publicly-available traffic datasets, showing that our model outperforms the baselines, especially in the long-term forecasting situation.

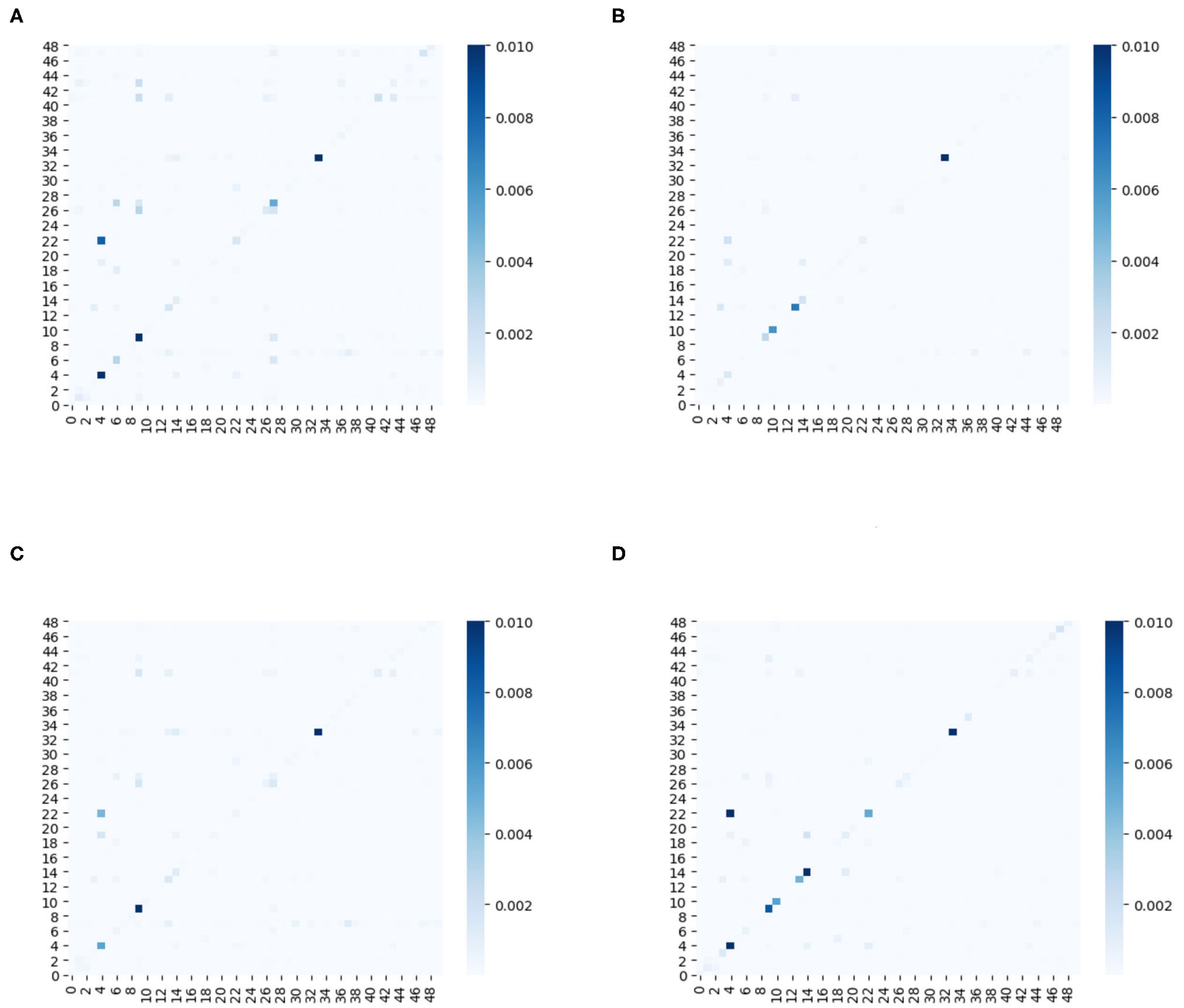


FIGURE 9 | (A–D) The heat maps of adjacency matrixes for the first 50 sensors at four different time slices.

Datasets and Preprocessing

We validate our model on two highway traffic datasets: PeMSD-08 and PeMSD-04 (12). The datasets were collected by a system called Caltrans Performance Measurement System (PeMS). On the highways of major metropolitan areas in California, more than 39,000 detectors were applied in this system to collect geographic and traffic information about the sensor locations. PeMSD-08 was collected in San Bernardino from July to August in 2016. This dataset contains 170 detectors on eight roads where the distance between any adjacent detectors is longer than 3.5 miles. The traffic data were aggregated every 5 min, so each detector contains 288 data points per day. PeMSD-04 is the traffic dataset collected by 307 detectors on 29 roads in San Francisco Bay Area. In this dataset, the time range is from January to February in 2018, and the time interval between two data points is 5 min, too.

To improve the model's efficiency and performance, it is necessary to normalize the input data and map their attribute values between$[0, 1]$. Min–max normalization was used to preprocess the datasets.

$$\mathrm{x}' = \frac{\mathrm{x} - \mathrm{x}_{\min}}{\mathrm{x}_{\max} - \mathrm{x}_{\min}} \tag{16}$$

where $x_{\min}$ denotes the minimum value of the input data, $x_{\max}$ denotes the maximum value of the input data, x is the observed data and x' is the normalized data.

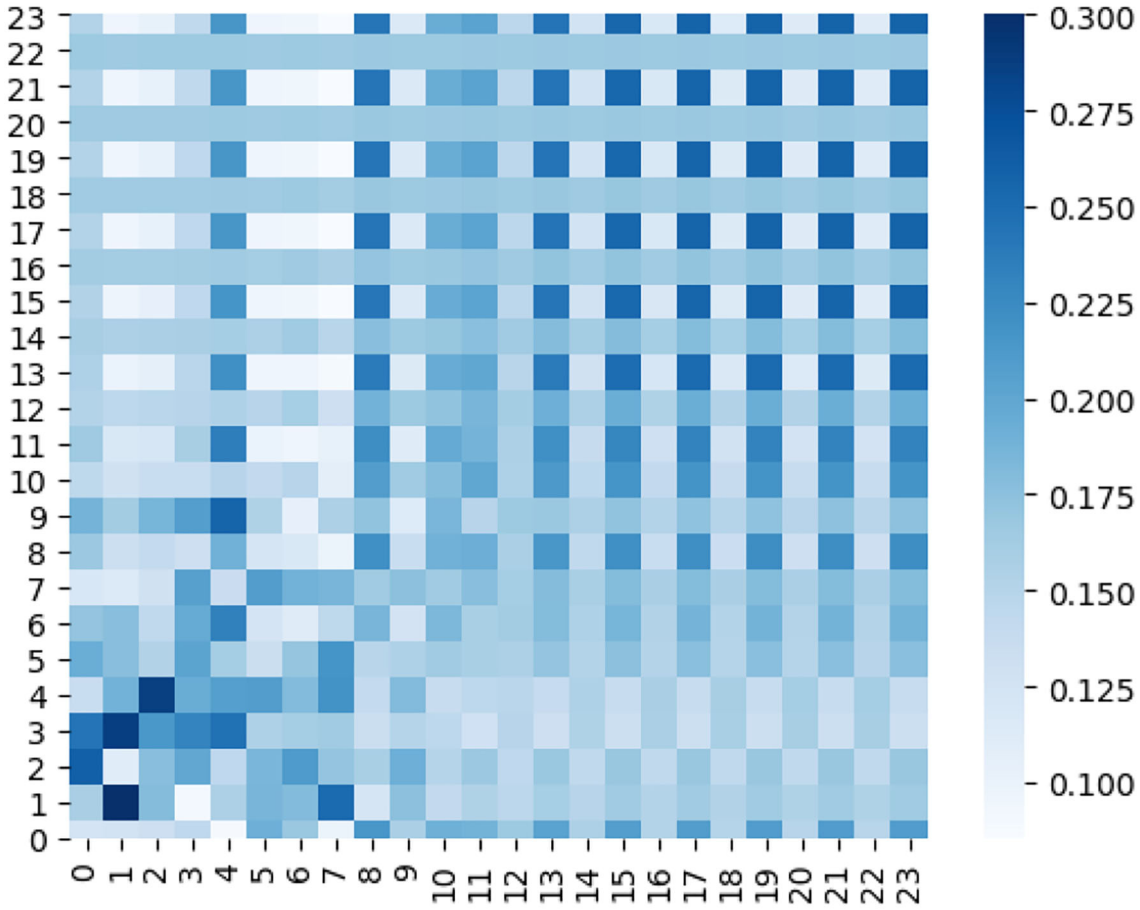


FIGURE 10 | Temporal ensemble attention coefficients of a certain sensor are visualized as a heat map.

We build the weighted adjacency matrix by road network distance:

$$A_{ij} = \begin{cases} 1 \text{ if } \mathrm{dist}(V_i, V_j) > E \\ 0 \quad \text{otherwise} \end{cases} \tag{17}$$

where A_{ij} represents the edge between node i and node j, $dist\left(V_i, V_j\right)$ represents the distance between node i and node j, and is the threshold to control the distribution and sparsity of matrix A.

Settings

We split the datasets in chronological order with the first 60% for training, next 20% for testing and the remaining 20% for validation. The proposed architecture is implemented by PyTorch (1.3.1 version) and trained on a computer with NVIDIA GeForce GTX 1050 GPU and Intel(R) i5-6500 CPU. The data input length is set to 24. Adam optimizer is chosen to optimize the parameters of our deep learning framework.

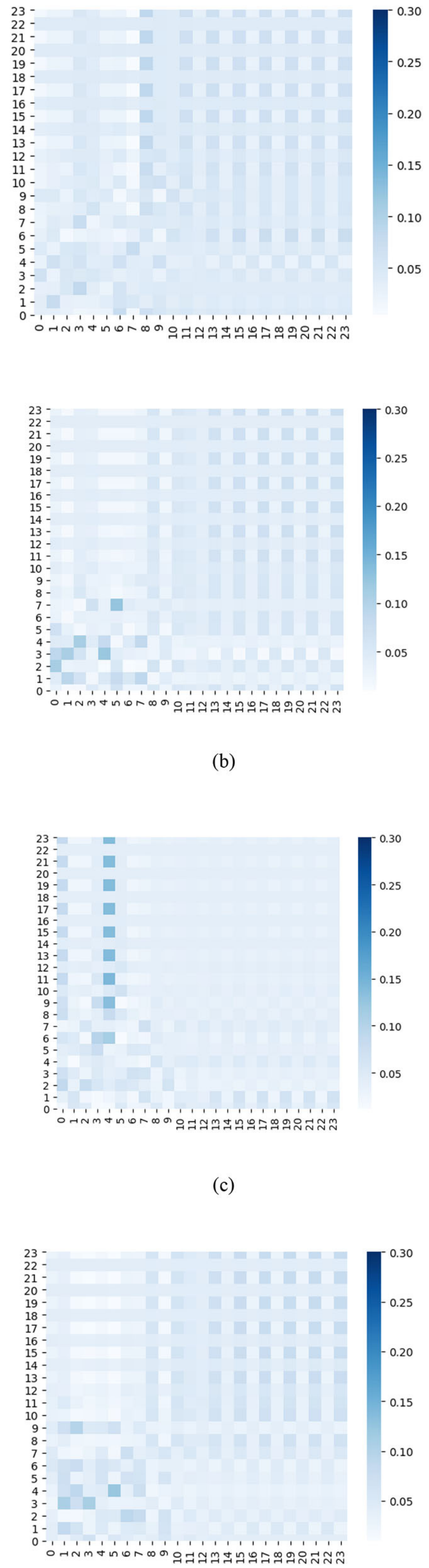
In our model, we set Chebyshev polynomial $K = 3$. As K continues to increase, the model's performance improves slightly with a much higher computational cost. The feature of GCN network's output is set to 64. The number of TCN's layers is set to four. The kernel size in TCN is set to three. We set the dilation $d = 2^l$, where $l \in \{0, 1, 2\}$ is the index number of layers in TCN. The number of the sub-attention block in temporal ensemble attention module is set to four. During the training phase, the batch size is eight and the learning rate is 0.001.

Two commonly used metrics: Mean Absolute Errors (MAE) and Root Mean Squared Errors (RMSE) were selected to evaluate the performance of different models.

$$\mathrm{MAE}\left(y, \hat{y}\right) = \frac{1}{Q}\sum_{i=1}^{Q}\left|y_i - \hat{y}_i\right| \tag{18}$$

$$\mathrm{RMSE}\left(y, \hat{y}\right) = \sqrt{\frac{1}{Q}\sum_{i=1}^{Q}\left(y_i - \hat{y}_i\right)^2} \tag{19}$$

where Q denotes the size of the testing dataset, y denotes the ground truth and ŷ denotes the predicted value.



(b)

(c)

FIGURE 11 | Temporal sub-attention coefficients of time slices are visualized as heat maps.

Baselines

We compare our model with several baseline models, including traditional machine learning methods (SVR, RF) and recently published state-of-the-art deep learning models (GRU, LSTM, Seq2Seq, ASTGCN) in traffic forecasting domains.

- ➢ SVR: Support Vector Regression, which uses support vector machine for the regression task. The model was implemented based on scikit-learn python package. The penalty term was set as 0.1, kernel type was set as "rbf," and the number of historical observation was set as 24.
- ➢ RF: Random Forest, which is made of many decision trees. The model was implemented based on scikit-learn python package. Estimators were set as 100, random state was set as 42, and the number of max features was set as six.
- ➢ GRU: Gated Recurrent Unit network, which is a special RNN model (47). The hidden size of GRU is set as 64.
- ➢ LSTM: Long Short Term Memory network, which is a special RNN model (48). The hidden size of LSTM is set as 64.
- ➢ Seq2Seq: Sequence to Sequence model, which is composed of the encoder and the decoder. The hidden size of both encoder and decoder GRUs are set as 64.
- ➢ ASTGCN: Attention Based Spatial-Temporal Graph Convolutional Network, which is composed of graph convolutional layers and normal 1-D convolutional (12).

Performance Comparisons and Analysis

Table 1 shows the performance of our model and other baselines for the traffic flow predictions in the next 15, 30, and 60 min on PeMSD-08 dataset.

It's obvious that our model obtains the best results in terms of all evaluation metrics. Specially, it is worth noting that the improvement of our model's performance than the second-best model (ASTGCN) increase as the forecasting horizon grows longer, which is shown in the last row of **Table 1**.

To further verify the superiority of our model in the long horizon forecasting situation, we compared the performances of our model and the other baselines for the next 60 min traffic flow prediction on both PeMSD-08 and PeMSD-04 datasets, which is shown by (**Table 2**). Obviously in the long-term traffic flow prediction situation, our model does perform better. We think this is because the TCN network in our model has stronger ability to capture long historical information. And the normal 1-D time convolution in ASTGCN model cannot get enough information from the past. On the other hand, **Table 3** shows two models' computation cost in the training. Our model costs less training time than the ASTGCN model. **Figure 7** shows the 60-min-ahead predicted values and the ground truth of a certain sensor in 2 days.

In order to verify the effectiveness of our model's ensemble-attention mechanism, we construct a new model by getting rid of the ensemble attention operation from our model. The reconstructed version of our model can be treated as a simple model stacked of the GCN layer and the TCN layer. We evaluated our model and the reconstructed version for the traffic flow predictions in the next 15, 30, 45, and 60 min on PeMSD-08 dataset. **Figure 8** shows the RMSE results of the comparison, and the model with the ensemble-attention mechanism outperforms the reconstructed version especially in the long-term forecasting situation. This means that the ensemble-attention mechanism does play a role in the traffic flow prediction.

Furthermore, we plot a series of heat maps with different color depth to visualize the learned ensemble-attention matrixes. Both the spatial ensemble attention and the temporal ensemble attention are obtained from the last unit of our model.

On the one hand, the first 50 nodes are picked out to show the spatial correlations among those nodes at four different time slices. **Figure 9** shows the heat maps of spatial ensemble attention matrixes at four different time slices. The X-axis and Y-axis denote the 50 detectors. The value of the pixel at point (x, y) is the coefficient of detector y to detector x. The color depth of pixel at point (x, y) indicates the degree of influence that the detector y exerts to the detector x. The pixel with a deep color indicates that the detector x is affected strongly by the detector y. From **Figure 9**, we can observe that the spatial ensemble attention mechanism does capture dynamic heterogeneous spatial correlations among nodes to a certain extent.

On the other hand, the heat map of temporal ensemble attention is shown by (**Figure 10**), and the temporal sub-attention blocks' heat maps are represented by (**Figure 11**). The X-axis and Y-axis denote the 24 time slices at a sensor node. The value of the pixel at point (x, y) is the coefficient of time slice y to time slice x. The pixel with a deep color indicates that time slice x is affected strongly by the time slice y. From **Figures 10**, **11**, we can observe that the temporal ensemble attention mechanism do capture some heterogeneous temporal correlations.

Based on the above analysis, the ensemble attention mechanism including the spatial part and temporal part do learn some beneficial information which enables the model to exploit the dynamic heterogeneous traffic patterns for traffic forecasting.

CONCLUSIONS AND FUTURE WORK

In this paper, a novel end-to-end deep learning framework is proposed for traffic flow predicting. The knowledge gained from our research can provide many valuable applications for vehicle emission warnings, improving urban traffic construction and studying the sources of air pollution. Each unit of the models mainly is composed by a spatial block and a temporal block. In the spatial block, we fuse GCN and spatial ensemble attention mechanism to capture global dynamic heterogeneous spatial patterns. In the temporal block, TCN and temporal ensemble attention mechanism are combined to capture non-stationary temporal patterns. The model is fed with a variety of explanatory variables including the historical traffic speed, the historical traffic density, the historical traffic flow and the graph of sensor network. Experiment results show that the forecasting accuracy of the proposed model is superior to existing models especially in the long-term predicting situation and the model can be trained faster than the main DL baselines. The ensemble attention mechanism is shown to be capable of capturing comprehensive dynamic heterogeneous spatial-temporal correlations of traffic series.

Actually, the urban traffic flow is affected by many external factors, such as weather and social events. In the future, we will fuse more external factors into our model to make the predictions of attributes related to traffic emissions more accurate.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

BY: supervision and writing–review and editing. AM: conceptualization, methodology, and writing-original draft. RF: methodology, programming, and writing-improvement. XS: data collection and data analysis. MZ and YY: investigation and data collection. All authors contributed to the article and approved the submitted version.

FUNDING

The work described in this paper was jointly supported by grants from National Natural Science Foundation of China (U1811463 and 51675077), the State Key Laboratory of Structural Analysis for Industrial Equipment (S18307).

REFERENCES

1. Institute HE. A critical review of the literature on emissions, exposure and health effects. *HEI special report.* (2010) 17:17.
2. Zhao ZY, Cao Y, Kang Y, Xu ZY. Prediction of spatiotemporal evolution of urban traffic emissions based on taxi trajectories. *Int J Autom Comput.* (2021) 18:219–32. doi: 10.1007/s11633-020-1271-y
3. Kelly FJ, Fussell JC. Air pollution and airway disease. *Clin Exp Allergy.* (2011) 41:1059–71. doi: 10.1111/j.1365-2222.2011.03776.x
4. Hoek G, Krishnan RM, Beelen R, Peters A, Ostro B, Brunekreef B. Long-term air pollution exposure and cardio-respiratory mortality. *Environ Health.* (2016) 12. doi: 10.1186/1476-069X-12-43
5. Adam M, Schikowski T, Carsin AE. Adult lung function and long-term air pollution exposure. escape: a multicentre cohort study and meta-analysis. *Eur Respir J.* (2014) 45:38–50. doi: 10.1183/09031936.00130014
6. May AA, Presto AA, Hennigan CJ, Nguyen NT, Gordon TD, Robinson AL. Gas-particle partitioning of primary organic aerosol emissions: (1) gasoline vehicle exhaust. *Atmos Environ.* (2013) 77:128–39. doi: 10.1016/j.atmosenv.2013.04.060
7. Xue H, Jiang S, and Liang B. A study on the model of traffic flow and vehicle exhaust emission. *Math Probl Eng.* (2013) 2013:1–6. doi: 10.1155/2013/736285
8. Ding AL, Zhao XM, Jiao LC. Traffic flow time series prediction based on statistics learning theory. Intelligent Transportation Systems, 2002. In: *Proceedings The IEEE 5th International Conference on IEEE.* New York, NY: IEEE(2002)
9. Boukerche A, Wang J. A performance modeling and analysis of a novel vehicular traffic flow prediction system using a hybrid machine learning-based model. *Ad Hoc Networks.* (2020) 106:102224. doi: 10.1016/j.adhoc.2020.102224
10. Wang X, Guan X, Cao J, Zhang N, Wu H. Forecast network-wide traffic states for multiple steps ahead: a deep learning approach considering dynamic non-local spatial correlation and non-stationary temporal dependency. *Transp Res Part C Emerg Technol.* (2020) 119:102763. doi: 10.1016/j.trc.2020.102763

11. Sun S, Zhang C, Zhang Y. Traffic flow forecasting using a spatio-temporal bayesian network predictor. *Lect Notes Comput Sci.* (2005) 3697:273–8. doi: 10.1007/11550907_43
12. Guo S, Lin Y, Feng N, Song C, Wan H. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Palo Alto, CF: Association for the Advancement of Artificial Intelligence (2019).
13. Yao H, Tang X, Wei H, Zheng G, Li Z. Revisiting spatial-temporal similarity: a deep learning framework for traffic prediction. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Palo Alto, CF: Association for the Advancement of Artificial Intelligence (2019).
14. Ahmed MS, Cook AR. Analysis of freeway traffic time-series data by using box-jenkins techniques. *Transp Res Rec.* (1979) 773:1–9.
15. Lee S, Fambro D, Fambro D. Application of subset autoregressive integrated moving average model for short-term freeway traffic volume forecasting. *Transp Res Rec.* (1999) 1678:179–88. doi: 10.3141/1678-22
16. Williams BM, Hoel LA. Modeling and forecasting vehicular traffic flow as a seasonal arima process: theoretical basis and empirical results. *J Transp Eng.* (2003) 129:664–72. doi: 10.1061/(ASCE)0733-947X
17. Guo J, Williams B. Real-time short-term traffic speed level forecasting and uncertainty quantification using layered kalman filters. *Transp Res Rec.* (2010) 2175:28–37. doi: 10.3141/2175-04
18. Chen S, Wang W, Ren G. A hybrid approach of traffic volume forecasting based on wavelet transform, neural network and Markov model. In: *IEEE International Conference on Systems Man and Cybernetics Conference Proceedings.* (2006) 1:393–8.
19. Chan KY, Dillon T, Singh SJ, Chang E. Neural-network-based models for short-term traffic flow forecasting using a hybrid exponential smoothing and levenberg–marquardt algorithm. *IEEE Transactions on Intelligent Transportation Systems*. (2012) 13:644–54. doi: 10.1109/TITS.2011.2174051
20. Yu B, Song X, Guan F, Yang Z, Yao B. k-nearest neighbor model for multiple-time-step prediction of short-term traffic condition. *J Transp Eng.* (2016) 142:4016018. doi: 10.1061/(ASCE)TE.1943-5436.0000816
21. Yao B, Chen C, Cao Q, Jin L, Zhang M, Zhu H, et al. Short-term traffic speed prediction for an urban corridor. *Computer-Aided Civil and Infrastructure Engineering.* (2017) 32:154–69. doi: 10.1111/mice.12221
22. Rice J, Vanzwet E. A simple and effective method for predicting travel times on freeways. *IEEE Transactions on Intelligent Transportation Systems.* (2004) 5–3:200–7. doi: 10.1109/TITS.2004.833765
23. Wu CH, Wei CC, Su DC, Chang MH, Ho JM. Travel time prediction with support vector regression. *IEEE Transactions on Intelligent Transportation Systems.* (2003) 5:276–81. doi: 10.1109/TITS.2004.837813
24. Xu D, Shi Y. A combined model of random forest and multilayer perceptron to forecast expressway traffic flow. In: *IEEE International Conference on Electronics Information and Emergency Communication.* Macau, China (2017).
25. Tian Y, Zhang K, Li J, Lin X, Yang B. Lstm-based traffic flow prediction with missing data. *Neurocomputing.* (2018) 318:297–305. doi: 10.1016/j.neucom.2018.08.067
26. Xu J, Rahmatizadeh R, Bölöni L, Turgut D. Real-time prediction of taxi demand using recurrent neural networks. *IEEE trans Intell Transp Syst*. (2018) 19:2572–81. doi: 10.1109/TITS.2017.2755684
27. Du S, Li T, Gong X, Yang Y, Horng SJ. Traffic flow forecasting based on hybrid deep learning framework. In: *2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE) PP*. Nanjing, China (2018).
28. Ke J, Zheng H, Yang H, Chen X. Short-term forecasting of passenger demand under on-demand ride services: a spatio-temporal deep learning approach. *Transp Res.* (2017) 85:591–608. doi: 10.1016/j.trc.2017.10.016
29. Shi X, Chen Z, Wang H, Yeung D-Y, Wong W-K, Woo W-C, et al. Convolutional LSTM network: a machine learning approach for precipitation nowcasting. *Adv Neural Inf Process Syst.* (2015) 8:802–10. doi: 10.1007/978-3-319-21233-3_6
30. Defferrard M, Bresson X, Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in Neural Information Processing Systems 29 (NIPS 2016).* (2016) 29:3844–52.
31. Li Z, Xiong G, Chen Y, Lv Y, Hu B, Zhu F. A hybrid deep learning approach with GCN and LSTM for traffic flow prediction*. In: *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. New York, NY: IEEE (2019).
32. Zhu H, Xie Y, He W, Sun C, Ma N. A novel traffic flow forecasting method based on rnn-gcn and brb. *J Adv Transp.* (2020) 2020:1–11. doi: 10.1155/2020/7586154
33. Kong X, Xing W, Wei X, Bao P, Lu W. Stgat: spatial-temporal graph attention networks for traffic flow forecasting. *IEEE Access.* (2020) 8:134363–72. doi: 10.1109/ACCESS.2020.3011186
34. Tang C, Sun J, Sun Y, Peng M, Gan N. A general traffic flow prediction approach based on spatial-temporal graph attention. *IEEE Access.* (2020) 8:153731–41. doi: 10.1109/ACCESS.2020.3018452
35. Cui Z, Henrickson K, Ke R, Wang YH. Traffic graph convolutional recurrent neural network: a deep learning framework for network-scale traffic learning and forecasting. *IEEE trans Intell Transp Syst.* (2019) 21:4883–94. doi: 10.1109/TITS.2019.2950416
36. Shuman DI, Narang SK, Frossard P, Ortega A, Vandergheynst P. The emerging field of signal processing on graphs: extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Process Mag.* (2013) 30:83–98. doi: 10.1109/MSP.2012.2235192
37. Sandryhaila A, Moura JMF. Discrete signal processing on graphs. *IEEE Trans. Signal Process.* (2013) 61:1644–56. doi: 10.1109/TSP.2013.2238935
38. Simonovsky M, Komodakis N. Dynamic edge-conditioned filters in convolutional neural networks on graphs. In: *30th Conference on Computer Vision and Pattern Recognition (CVPR 2017).* Honolulu, USA, (2017).
39. Aerde MV, Rakha H. Multivariate calibration of single regime speed-flow-density relationships [road traffic management]. In: *1995 Vehicle Navigation and Information Systems Conference*. New York, NY: IEEE (1995).
40. Spyropoulou I. Modelling a signal controlled traffic stream using cellular automata. *Transp Res Part C Emerg Technol.* (2007) 15:175–90. doi: 10.1016/j.trc.2007.04.001
41. Chen D, Zhang J, Tang S, Wang AJ. Freeway traffic stream modeling based on principal curves and its analysis. *IEEE trans Intell Transp Syst.* (2004) 5:246–58. doi: 10.1109/TITS.2004.838226
42. Xu Z, Wang Y, Wang G, Li X, Zhao X. Trajectory optimization for a connected automated traffic stream: comparison between an exact model and fast heuristics. *IEEE trans Intell Transp Syst.* (2020) 1–10. doi: 10.1109/TITS.2020.2978382
43. Rakha H, Crowther B. Comparison of greenshields, pipes, and van aerde car-following and traffic stream models. *Transp Res Rec.* (2002) 1802:248–62. doi: 10.3141/1802-28
44. Laha A, Raykar V. An Empirical Evaluation of various Deep Learning Architectures for Bi-Sequence Classification Tasks. *arXiv [preprint]*, arXiv:1607, 04853 (2016).
45. Bai S, Kolter JZ, Koltun V. Trellis Networks for Sequence Modeling. *arXiv [preprint]*, arXiv:1803, 01271 (2018).
46. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN. Attention is all you need. In: *2017 Neural Information Processing Systems (NIPS)* California, CF: Neural Information Processing Systems (2017).
47. Chung J, Gulcehre C, Cho KH. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. CoRR, abs/1412:3555 British Virgin Islands: OAlib (2014).
48. Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput.* (1997) 9:1735–80. doi: 10.1162/neco.1997.9.8.1735

Conflict of Interest: XS was employed by CIECC Overseas Consulting Co., Ltd.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.