10-20-2022

Convolutional Long-Short Term Memory Network with Multi-Head Attention Mechanism for Traffic Flow Prediction

Yupeng Wei
San Jose State University, yupeng.wei@sjsu.edu

Hongrui Liu
San Jose State University, hongrui.liu@sjsu.edu

Recommended Citation

Yupeng Wei and Hongrui Liu. "Convolutional Long-Short Term Memory Network with Multi-Head Attention Mechanism for Traffic Flow Prediction" *Sensors (Basel, Switzerland)* (2022). https://doi.org/10.3390/s22207994

sensors

MDPI

Article

Convolutional Long-Short Term Memory Network with Multi-Head Attention Mechanism for Traffic Flow Prediction

Yupeng Wei * and Hongrui Liu

Department of Industrial and Systems Engineering, San Jose State University, San Jose, CA 95192, USA
* Correspondence: yupeng.wei@sjsu.edu

Abstract: Accurate predictive modeling of traffic flow is critically important as it allows transportation users to make wise decisions to circumvent traffic congestion regions. The advanced development of sensing technology makes big data more affordable and accessible, meaning that data-driven methods have been increasingly adopted for traffic flow prediction. Although numerous data-driven methods have been introduced for traffic flow predictions, existing data-driven methods cannot consider the correlation of the extracted high-dimensional features and cannot use the most relevant part of the traffic flow data to make predictions. To address these issues, this work proposes a decoder convolutional LSTM network, where the convolutional operation is used to consider the correlation of the high-dimensional features, and the LSTM network is used to consider the temporal correlation of traffic flow data. Moreover, the multi-head attention mechanism is introduced to use the most relevant portion of the traffic data to make predictions so that the prediction performance can be improved. A traffic flow dataset collected from the Caltrans Performance Measurement System (PeMS) database is used to demonstrate the effectiveness of the proposed method.

Keywords: traffic flow prediction; deep learning; convolutional LSTM; attention mechanism

Citation: Wei, Y.; Liu, H. Convolutional Long-Short Term Memory Network with Multi-Head Attention Mechanism for Traffic Flow Prediction. *Sensors* **2022**, *22*, 7994. https://doi.org/10.3390/s22207994

Academic Editors: Shyan-Ming Yuan, Zeng-Wei Hong and Wai-Khuen Cheng

Received: 9 October 2022
Accepted: 18 October 2022
Published: 20 October 2022

1. Introduction

Traffic congestion results in reduced efficiency of transportation infrastructure, increased traveling time, and a waste of energy fuel [1–3]. According to a report by Nationwide, 1.9 billion gallons of fuel are wasted every year as a result of traffic congestion [4]. Traffic congestion could be induced by numerous factors, such as bottlenecks, traffic accidents, and severe weather conditions. To address the issue of traffic congestion, traffic flow prediction has gained much attention in the recent decade. Accurate predictive modeling of traffic flow is critically important as it allows transportation users to make wise decisions to circumvent traffic congestion regions [5]. Therefore, commuter and shipment activities could be effectively scheduled to increase moving efficiency. Moreover, accurate predictive modeling of traffic flow can also assist in reducing carbon emissions and traffic incident possibilities.

The advanced development of sensing technology makes big data more affordable and accessible, and thus, data-driven methods have been increasingly adopted for the predictive modeling of traffic flow. Data-driven methods can be classified into two categories: machine learning methods and deep learning methods [6–10]. In comparison with machine learning methods, deep learning methods have gained more attention from both academia and industry in traffic flow predictions due to their extraordinary prediction fidelity and robustness. Among these deep learning methods, artificial neural networks (ANNs) and autoencoder-based methods have been widely used for traffic flow predictions as these methods are capable of decomposing the original traffic flow data into features located at a higher dimensional feature space, and these high-dimensional features can reveal the latent information in the traffic flow data. However, there are two primary issues for ANNs and autoencoders: (1) they can not take the temporal correlation of traffic flow data into account;

(2) they can not consider the correlation of the extracted high-dimensional features. To consider the temporal correlation of traffic flow data, deep learning methods with recurrent characteristics are adopted, such as long short-term memory (LSTM), recurrent neural network (RNN), and gated recurrent unit (GRU). While these deep learning methods with recurrent characteristics are promising, they are not able to use the most relevant part of the traffic flow data to make predictions, which leads to a higher prediction time and a worse prediction accuracy. To address these issues, this work introduces a novel deep learning-based framework to consider the temporal correlation of traffic flow data, the correlation of the extracted high-dimensional features, and the most relevant part of the traffic flow data to make predictions in a unified manner. More specifically, a decoder network is firstly proposed to decompose the traffic flow data into high-dimensional features. Second, a convolutional LSTM network is introduced to simultaneously consider the correlation of the decomposed high-dimensional features and the temporal correlation of traffic flow data, where the convolutional operation is used to consider the correlation of the high-dimensional features, and the LSTM network is used to consider the temporal correlation of traffic flow data. Next, the multi-head attention mechanism is introduced to use the most relevant portion of the traffic data to make predictions so that the prediction performance can be improved. The primary contribution of this work can be summarized as follows:

- A decoder network is introduced to decompose the original traffic flow data into features located at a higher-dimensional feature space.
- A convolutional LSTM network is introduced to consider the correlation of the high dimensional features and the temporal correlation of traffic flow data.
- A multi-head attention mechanism is introduced to use the most relevant portion of the traffic data to make predictions so that the prediction performance can be improved.

The remainder of this paper is organized as follows. Section 2 reviews data-driven methods reported in the literature for traffic flow predictions. Section 3 introduces the proposed deep learning model. Section 4 demonstrates the effectiveness of the proposed method utilizing the traffic flow data from the Caltrans Performance Measurement System (PeMS) database. Section 5 concludes this research work and directs future work.

2. Literature Review

In the context of traffic flow predictions, data-driven methods can be classified into two categories: machine learning [11–13] and deep learning methods [14,15]. These machine learning methods include support vector regression [16], random forest [17], Gaussian process [18], Bayesian models [19], and so on. For example, Tang et al. [20] combined the support vector machine method with multiple denoising mechanisms to predict the traffic flow. A dataset collected by the real-time detectors located in the city of Minneapolis was used to evaluate the performance of the proposed methods. The simulation results have shown that the denoising mechanisms could boost the performance of the support vector machine. Zhang et al. [21] introduced a hybrid framework based upon support vector regression to predict the traffic flow, where the random forest method was implemented for feature selections, and the genetic algorithm was adopted to determine the model hyperparameters. The simulation results have shown that the proposed methodology enables better prediction accuracy. Xu et al. [22] introduced a scalable Gaussian process model for large-scale traffic flow predictions. The proposed model combined the Gaussian process with alternative directional methods for paralleling and optimizing hyperparameters during the training process. Wang et al. [23] presented a vicinity Gaussian process method for short-term traffic flow prediction under the conditions of missing data with measuring errors. In the proposed model, a directed graph was constructed based on the traffic network, a dissimilarity matrix and a proper cost function were selected to boost the prediction performance. Zhu et al. [24] introduced a linear conditional Gaussian process method, where temporal and spatial correlations of traffic flow were taken into account. A simulated traffic dataset was adopted to evaluate the effectiveness of the Gaussian process

method, and simulation results have shown that the utilization of both spatial and temporal data can dramatically boost prediction accuracy. Li et al. [25] presented a Bayesian network to tackle the node selection challenge in traffic flow prediction. Experimental results have shown that the proposed directed correlation-based Bayesian network method results in a sparse model and better performance in traffic flow prediction.

With the advanced improvement of computational power, deep learning methods are increasingly adopted in traffic flow prediction due to their extraordinary performance. These deep learning methods include LSTM [26,27], gated recurrent neural network (GRU) [28,29], recurrent neural network (RNN) [30,31], graph neural network (GNN) [32–34], and so on. For instance, Tian et al. [35] introduced LSTM-based predictive modeling of traffic flow, where a smoothing function was implemented to deal with the missing data points, and the LSTM was used to capture the prediction residual. Two traffic flow datasets were used to evaluate the performance of the proposed methodology, and the results have shown that the smoothing function can boost the performance of the predictive model. Dai et al. [36] integrated the spatial-temporal analysis with a GRU network to forecast the traffic flow in a short time interval. In the proposed method, the GRU model was applied to process the spatial-temporal features extracted from the collected traffic data. The simulation results have shown that the GRU outperforms the convolutional neural network (CNN) in both prediction accuracy and robustness. Zhene et al. [37] combined the CNN with RNN for urban traffic flow predictions, where CNN was adopted to extract attributes from traffic flow data and RNN was implemented to make predictions. In comparison with the traditional RNN, the proposed RNN was able to process multiple temporal features simultaneously. The experimental results have demonstrated that online traffic flow prediction could be achieved with high precision by using the proposed methodology. Luo et al. [38] introduced a k-nearest neighbor-based (KNN) LSTM method to extract temporal and spatial correlations, where KNN was utilized to capture spatial correlations and LSTM was adopted to further extract temporal correlations. A dataset provided by the University of Minnesota Duluth Data center was utilized to demonstrate the effectiveness of the proposed methods, and the results have indicated that the proposed method outperforms the auto-regressive integrated moving average and wavelet neural network in terms of prediction accuracy. Zhu et al. [39] integrated the GNN with RNN to extract the spatial and temporal correlations of traffic data. The belief rule-based algorithm was adopted for data fusion, and the fused traffic data were fed into the proposed methodology for traffic flow prediction. Yu et al. [40] presented a novel GNN methodology to predict the traffic flow, in which a weighted undirected graph was utilized to differentiate the density of connected roads. A simulation model was introduced to simulate the traffic propagation, and the simulation results were considered in the GNN model for online traffic flow prediction. The simulation results have shown that the proposed GNN outperforms the traditional GNN in traffic flow predictions. More details about applying GNN for traffic flow predictions can be found in [41].

While numerous data-driven methods have been studied to predict traffic flow under various conditions, some issues still exist with these methods. The existing data-driven methods can not consider the correlation of the extracted high-dimensional features and can not use the most relevant part of the traffic flow data to make predictions, which leads to a higher prediction time and a worse prediction accuracy. To deal with these issues, this work proposes a decoder convolutional LSTM network to simultaneously consider the correlation of the decomposed high-dimensional features and the temporal correlation of traffic flow data, where the convolutional operation is used to consider the correlation of the high-dimensional features, and the LSTM network is used to consider the temporal correlation of traffic flow data. Moreover, a multi-head attention mechanism is introduced to use the most relevant portion of the traffic data to make predictions so that the prediction performance can be improved.

3. Convolutional LSTM with Multi-Head Attention Mechanism

This section introduces the convolutional LSTM with a multi-head attention mechanism. Figure 1 shows the framework of the proposed deep learning approach. First, a moving window with a fixed window size is utilized to split raw traffic flow into historical traffic flow as features and future traffic flow as labels. The historical traffic flow is fed into a decoder network to be decomposed into multiple time-series signals. The decomposed signals are fed into the convolutional LSTM network to consider the correlation of the decomposed high dimensional features and the temporal correlation of traffic flow data. The outputs of the convolutional LSTM are transited to the multi-head attention model for traffic flow prediction. Next, the prediction loss is calculated based on the future traffic flow and predicted traffic flow, and the backpropagation algorithm is adopted to train the proposed method. More details of the proposed deep learning approach are provided in the following subsections.

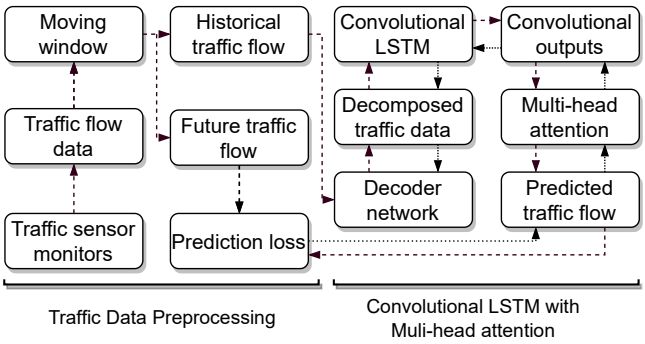


Figure 1. The framework of the convolutional LSTM with a multi-head attention mechanism for traffic flow prediction.

3.1. Decoder Network for Traffic Data Decomposition

The initial step of the proposed method is to decompose the traffic flow so that the most useful latent information can be reflected and the data can be better analyzed. To decompose the traffic flow data, this research uses a decoder network that stacks multiple fully connected layers. The output of the decoder network can be written as Equation (1),

$$\mathbf{D}_{i,L} = f_L \dots [f_l \dots [f_2[f_1(\mathbf{X}_i)]]] \tag{1}$$

where $\mathbf{X}_i \in \mathbb{R}^{1\times T}$ represent the traffic flow data for data sample i; L refers to the total number of stacked fully connected layers in the decoder network; $\mathbf{D}_{i,L} \in \mathbb{R}^{m\times T}$ refers to the output of the decoder network for data sample i; m represents the number of hidden nodes in the fully connected layers of the decoder network; T represents the length of the historical traffic flow; and $f_l(\cdot)$ can be given by Equation (2).

$$f_l(\cdot) := Relu(\mathbf{W}_l \cdot \mathbf{D}_{i,l-1} + \mathbf{b}_l) \tag{2}$$

In Equation (2), $Relu$ represents the rectified linear unit activation function; $\mathbf{W}_l$ refers to the kernel weight matrix at the l-th fully connected layer in the decoder network; $\mathbf{D}_{i,l-1}$ represents the output of the $l-1$-th fully connected layer for data sample i; and $\mathbf{b}_l$ represents the bias weight matrix at the l-th fully connected layer. Next, the output $\mathbf{D}_{i,L}$ of the decoder network is fed into the convolutional LSTM network to consider the correlation of the decomposed high-dimensional features and the temporal correlation of traffic flow data.

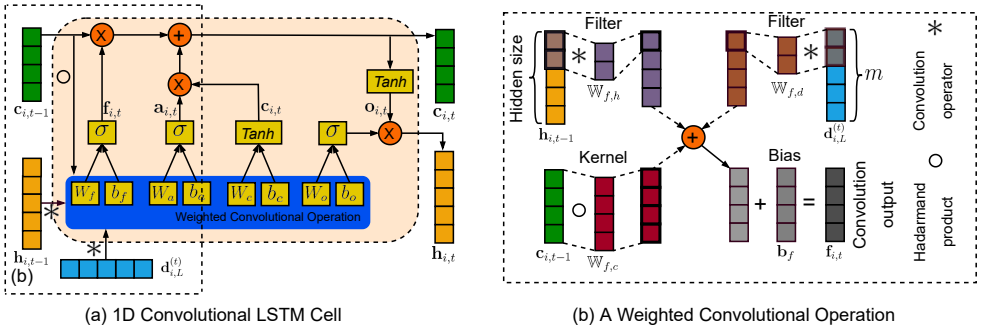
3.2. Convolutional LSTM Cell

The traditional LSTM is capable of considering the temporal correlation of traffic flow data. However, the traditional LSTM fails to consider the correlation of the decomposed

high-dimensional features. To address this issue, this research aims to introduce the convolutional LSTM cell that incorporates a convolutional operation into the traditional LSTM cell so that both the temporal correlation of traffic flow data and the correlation of the decomposed high-dimensional features can be considered in a unified manner [42]. Figure 2 shows the framework of the convolutional LSTM cell. In the convolutional LSTM cell, the output vector $\mathbf{d}_{i,L}^{(t)}$ of the decoder network at time t and the hidden state $\mathbf{h}_{i,t-1}$ of the one-dimensional convolutional LSTM cell at the prior time point $t-1$ are fed into the one-dimensional convolutional LSTM cell to perform the weighted convolutional operations. Such convolutional operations can consider the correlation of the decomposed high dimensional features $\mathbf{D}_{i,L}$. The recurrent usage of the convolutional LSTM cell can extract temporal correlations, and the output of this cell can be written as Equation (3),

$$\begin{aligned}
\mathbf{f}_{i,t} &= \sigma(C_{i,f} + \mathbb{W}_{f,c} \circ \mathbf{c}_{i,t-1} + \mathbf{b}_f) \\
\mathbf{a}_{i,t} &= \sigma(C_{i,a} + \mathbb{W}_{a,c} \circ \mathbf{c}_{i,t-1} + \mathbf{b}_a) \\
\mathbf{c}_{i,t} &= \mathbf{f}_{i,t} \circ \mathbf{c}_{i,t-1} + \mathbf{a}_{i,t} \circ Tanh(C_{i,c} + \mathbf{b}_c) \\
\mathbf{o}_{i,t} &= \sigma(C_{i,o} + \mathbb{W}_{o,c} \circ \mathbf{c}_{i,t} + \mathbf{b}_o) \\
\mathbf{h}_{i,t} &= \mathbf{o}_{i,t} \circ \sigma(\mathbf{c}_{i,t})
\end{aligned} \tag{3}$$

where $\mathbf{f}_{i,t}, \mathbf{a}_{i,t}, \mathbf{c}_{i,t}, \mathbf{o}_{i,t}$, respectively, refer to the outputs of the forget gate, input gate, memory cell, and output gate; $\mathbb{W}_{f,c}, \mathbb{W}_{a,c}, \mathbb{W}_{o,c}$ represent the trainable matrices for the forget gate, input gate, and output gate, respectively; $\mathbf{b}_f, \mathbf{b}_a, \mathbf{b}_c, \mathbf{b}_o$ represent the bias vectors for the forget gate, input gate, memory cell, and output gate; σ refers to the sigmoid function; *Tanh* refers to the hyperbolic tangent function.



(a) 1D Convolutional LSTM Cell (b) A Weighted Convolutional Operation

Figure 2. The framework of the one-dimensional convolutional LSTM cell with weighted convolutional operations, where (**a**) is the 1D convolutional LSTM cell and (**b**) gives an example of the weighted convolutional operation.

Moreover, $C_{i,f}, C_{i,a}, C_{i,c}, C_{i,o}$, respectively, refer to the outputs of the convolutional operations at the forget gate, input gate, memory cell, and output gate. These convolutional outputs can be written as Equation (4), where $*$ refers to the convolutional multiplication; $\mathbb{W}_{f,d}$ and $\mathbb{W}_{f,h}$ refer to the kernel matrices of the convolutional operations at the forget gate; $\mathbb{W}_{a,d}$ and $\mathbb{W}_{a,h}$ are the kernel matrices of the convolutional operations at the input gate; $\mathbb{W}_{c,d}$ and $\mathbb{W}_{c,h}$ represent the kernel matrices of the convolutional operations in the memory cell; and $\mathbb{W}_{o,d}$ and $\mathbb{W}_{o,h}$ represent the kernel matrices of the convolutional operations at the output gate.

$$\begin{cases}
C_{i,f} = \mathbb{W}_{f,d} * \mathbf{d}_{i,L}^{(t)} + \mathbb{W}_{f,h} * \mathbf{h}_{i,t-1} \\
C_{i,a} = \mathbb{W}_{a,d} * \mathbf{d}_{i,L}^{(t)} + \mathbb{W}_{a,h} * \mathbf{h}_{i,t-1} \\
C_{i,c} = \mathbb{W}_{c,d} * \mathbf{d}_{i,L}^{(t)} + \mathbb{W}_{c,h} * \mathbf{h}_{i,t-1} \\
C_{i,o} = \mathbb{W}_{o,d} * \mathbf{d}_{i,L}^{(t)} + \mathbb{W}_{o,h} * \mathbf{h}_{i,t-1}
\end{cases} \tag{4}$$

In summary, the convolutional LSTM cell integrates the convolutional operations with the traditional LSTM cell, where the convolutional operations are adopted to consider the

correlation of the decomposed high-dimensional features $\mathbf{D}_{i,L}$ and the traditional LSTM cell is utilized to extract the temporal correlations of traffic flow data. The integration of the convolutional operation with the traditional LSTM cell allows the neural network to consider both the correlation of the decomposed high-dimensional features and the temporal correlation of traffic flow data. Next, the hidden outputs, $\mathbf{h}_{i,t}$ for all t, of the convolutional LSTM cell are fed into the multi-head attention mechanism for the final prediction.

3.3. Multi-Head Attention Model

In the recent decade, the attention mechanism [43,44] has been introduced to deal with time series as it is capable of using the most relevant proportion of a time series to make predictions. The primary theory of the attention mechanism is simulating the data retrieval process in the data management system. To retrieve data, a query should be inserted into a data management system. If the query is matched with a key, the value associated with the key will be retrieved. Equation (5) shows the construction process of queries Q_i, keys K_i, and values V_i for traffic flow predictions.

$$(\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V) \cdot \mathbf{H}_i = (Q_i, K_i, V_i) \tag{5}$$

In Equation (5), $\mathbf{H}_i$ represents the hidden outputs of the convolutional LSTM network for data sample i, and $\mathbf{H}_i$ can be written as $\mathbf{H}_i = (\mathbf{h}_{i,1}, \ldots, \mathbf{h}_{i,t}, \ldots, \mathbf{h}_{i,T})$; and $\mathbf{W}_Q \in \mathbb{R}^{r \times T}$, $\mathbf{W}_K \in \mathbb{R}^{r \times T}$, $\mathbf{W}_V \in \mathbb{R}^{r \times T}$ are trainable weight matrices. To use the most relevant portion of the values V, the attention vector $\mathbf{a}$ should be obtained by using Equation (6), where $SoftMax$ is the normalized exponential function.

$$\mathbf{a} = SoftMax(Q_i \cdot K_i' / \sqrt{T}) \tag{6}$$

To retrieve the most relevant part of the values V, the attention vector is multiplied by the value matrix, which can be written as $\mathbf{O}_i = \mathbf{a}V_i$.

The multi-head attention mechanism stacks the multiple attention model [45,46]. Figure 3 presents the framework of the multi-head attention model for traffic flow prediction. The attention vector of the multi-head attention mechanism can be written as $\mathbf{a}_h = SoftMax(\mathbf{W}_Q^{(h)}\mathbf{H}_i \cdot (\mathbf{W}_K^{(h)}\mathbf{H}_i)' / \sqrt{T})$, where $\mathbf{W}_Q^{(h)}, \mathbf{W}_K^{(h)}, \mathbf{W}_V^{(h)}$ are trainable weight matrices of the h-th attention model; and $\mathbf{a}_h$ is the attention vector of the h-th attention model. The output of the h-th attention model is written as $\mathbf{O}_{i,h} = \mathbf{a}_h(\mathbf{W}_V^{(h)}\mathbf{H}_i)$.

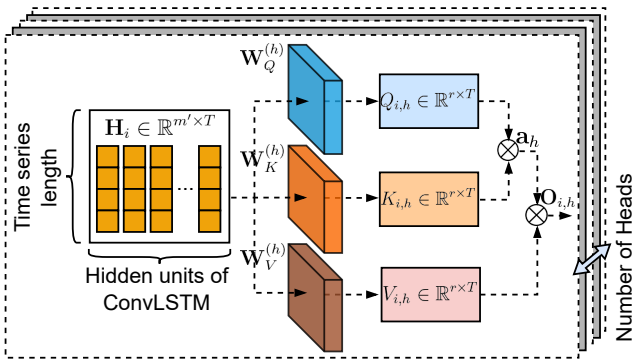


Figure 3. The framework of the multi-head attention model for traffic flow prediction.

Next, the output of all attention models is concatenated, which can be written as Equation (7), where H is the number of attention models and has been stacked in the multi-head attention model.

$$\mathbf{C}_i = Concat\{\mathbf{O}_{i,1}, \ldots, \mathbf{O}_{i,h}, \ldots, \mathbf{O}_{i,H}\} \tag{7}$$

Next, the concatenated output **C** is fed into a fully connected layer for final predictions. The training loss of the traffic flow prediction is written as Equation (8), where N refers to the total amount of data samples; $y_{i,j}$ is the true traffic flow for sample i at time j; and $\hat{y}_{i,j}$ is the predicted traffic flow for sample i at time j.

$$L = \frac{1}{N \times T} \sum_{i=1}^{N} \sum_{j=1}^{T} (y_{i,j} - \hat{y}_{i,j})^2 \tag{8}$$

The backpropagation algorithm is utilized for training the proposed deep learning model. Table 1 presents the training process of the proposed method. First, the weight matrices in the deep learning model are randomly initialized, the traffic flow data and labels are prepared, and the learning rate is initialized. Next, the traffic flow data $\mathbf{X}_i$ for data sample i are fed into the decoder network to decompose the traffic flow data into multiple parts. The output $\mathbf{D}_{i,L}$ of the decoder network is fed into the convolutional LSTM layer to extract temporal and spatial correlations, and the output of this layer is $\mathbf{H}_i$. Next, $\mathbf{H}_i$ is fed into the multi-head attention model to use the most relevant portion of the features extracted by the convolutional LSTM layer. The output of the multi-head attention model $\mathbf{C}_i$ is fed into the fully connected layers for traffic flow predictions, and the trainable weight matrices are updated in each training iteration.

Table 1. The pseudo-code to train the proposed deep learning model for traffic flow predictions.

1. Initialize trainable weight matrices
2. Prepare the traffic flow data $\mathbf{X}_i$ and the traffic flow labels $y_{i,j}, \forall i, j$
3. Initialize the learning rate
4. While iteration = 1,...,I, repeat
  4.1. While $l = 1, \ldots, L$, repeat
    $\mathbf{D}_{i,l} = Relu(\mathbf{W}_l \cdot \mathbf{D}_{i,l-1} + \mathbf{b}_l)$, $\mathbf{D}_{i,l} = \mathbf{X}_i$ if $l = 1$
  4.2. End iteration
  4.3. Feed $\mathbf{D}_{i,L}$ into the convolutional LSTM layer to obtain $\mathbf{H}_i$
  4.4. While $h = 1, \ldots, H$, repeat
    Obtain attention vector $\mathbf{a}_h \leftarrow SoftMax(Q_{i,h} \cdot K'_{i,h} / \sqrt{T})$
    Obtain attention model's output $\mathbf{O}_{i,h} \leftarrow \mathbf{a}_h \cdot V_{i,h}$
  4.5. End iteration
  4.6. Obtain $\mathbf{C}_i \leftarrow Concat\{\mathbf{O}_{i,1}, \ldots, \mathbf{O}_{i,h}, \ldots, \mathbf{O}_{i,H}\}$
  4.7. Feed $\mathbf{C}_i$ to FC layers
  4.8. Update weight matrices in fully connected layers
  4.9. Update weight matrices in the multi-head attention layer
  4.10. Update weight matrices in convolutional LSTM layer
  4.11. Update weight matrices in the decoder network
5. End iteration

4. Case Study

In this section, a real-world traffic flow dataset was used to demonstrate the effectiveness of the proposed deep learning approach. The following subsections provide dataset descriptions, evaluation metrics, model architecture, and prediction results.

4.1. Dataset Description

Traffic flow data collected by the Caltrans Performance Measurement System (PeMS) was utilized to demonstrate the effectiveness of the proposed methodology. The dataset was collected in real-time from over 40,000 unique detectors located on the freeway in the state of California [47]. The collected dataset aggregated hourly traffic flow data obtained from the corresponding detection station. In this study, we used two cases to demonstrate the effectiveness of the proposed method. The first case used the traffic flow data collected from January to March in the year 2022 located at the I5-North freeway, where the post-mile range is from 495.73 to 621.42 in the state of California. The second case used the traffic flow data collected from February to April in the year 2022 located at the I5-North freeway,

where the post-mile range is from 495.73 to 621.42 in the state of California. The post-mile refers to the range of routes that move through individual counties in the state of California. For both two cases, the data for the first two months were used to train the proposed deep learning model, and the remaining month was used to test the proposed model. Figure 4 highlights the range of the post-mile 495.73 to 621.42 at the freeway I5-North. To avoid loss of generality, both training and test data were standardized. In this work, we use the data rescaling method to standardize all data to guarantee that both vehicle miles traveled (VMT) and vehicle hours traveled (VHT) are on the same scale. The data rescaling method refers to multiplying each data point by a constant factor, where the factors for VMT and VHT are 10^{-5} and 10^{-3}, respectively.

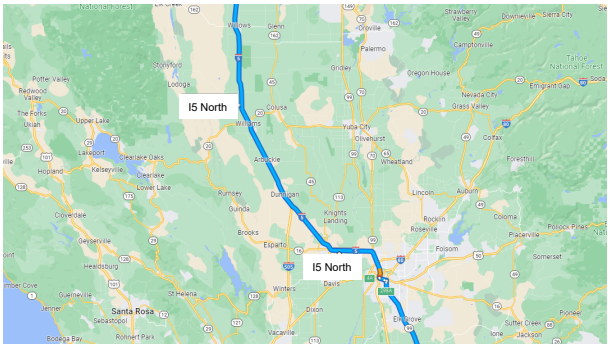


Figure 4. The post-mile ranges from 495.73 to 621.42 located at the freeway I5-North.

4.2. Evaluation Metric

To evaluate the performance of the proposed methodology, this study adopts the root mean squared error (RMSE) and mean absolute error (MAE). The RMSE and MAE can be defined by using Equation (9), where N is the total amount of data samples; $y_{i,j}$ refers to the true traffic flow for the sample i at time j; and $\hat{y}_{i,j}$ represents the predicted traffic flow for the sample i at time j.

$$\begin{aligned} RMSE &= \left(\frac{1}{N \times T}\sum_{i=1}^{N}\sum_{j=1}^{T}(y_{i,j} - \hat{y}_{i,j})^2\right)^{1/2} \\ MAE &= \frac{1}{N \times T}\sum_{i=1}^{N}\sum_{j=1}^{T}|y_{i,j} - \hat{y}_{i,j}| \end{aligned} \tag{9}$$

4.3. Model Architecture and Hyperparameters

In this case study, we use three tasks to evaluate the prediction performance of the proposed deep learning model for both two cases. These tasks include the next 1st-hour traffic flow prediction (first task), the next 5th-hour traffic flow prediction (second task), and the next 10th-hour traffic flow prediction (third task). The next nth-hour traffic flow prediction refers to using the past 24 h traffic flow data to predict the traffic flow in the $24 + n$ h. Tables 2–4 show the model architecture and hyperparameters used in this case study for three tasks. For these three tasks under two cases, we use the batch size of 100 and utilize the past 24 h traffic flow data to make predictions in each batch. We also use the filter size of 2 in the first task and use the filter size of 10 in the remaining two tasks. Moreover, the number of hidden nodes in the decoder network is 100.

Table 2. The model architecture and hyperparameters used for the next 1st-hour traffic flow prediction task.

No. of Layers	Descriptions	Output Dimensions
1	Input layer	100 × 24 × 1
2	FC layer	100 × 24 × 100
3	Convolutional LSTM	100 × 24 × 99
4	Multi-head attention	100 × 24 × 99
5	Flatten layer	100 × 2376
6	Dense layer	100 × 1

Table 3. The model architecture and hyperparameters used for the next 5th-hour traffic flow prediction task.

No. of Layers	Descriptions	Output Dimensions
1	Input layer	100 × 24 × 1
2	FC layer	100 × 24 × 100
3	FC layer	100 × 24 × 100
4	FC layer	100 × 24 × 100
5	Convolutional LSTM	100 × 24 × 91
6	Multi-head attention	100 × 24 × 91
7	Flatten layer	100 × 2184
8	Dense layer	100 × 1

Table 4. The model architecture and hyperparameters used for the next 10th-hour traffic flow prediction task.

No. of Layers	Descriptions	Output Dimensions
1	Input layer	100 × 24 × 1
2	FC layer	100 × 24 × 100
3	FC layer	100 × 24 × 100
4	FC layer	100 × 24 × 100
5	FC layer	100 × 24 × 100
6	FC layer	100 × 24 × 100
7	Convolutional LSTM	100 × 24 × 91
8	Multi-head attention	100 × 24 × 91
9	Flatten layer	100 × 2184
10	Dense layer	100 × 1

4.4. Traffic Flow Prediction Results for the First Case

Figure 5 shows the traffic flow prediction results for three different tasks under the first case, where VMT refers to vehicle miles traveled, and VHT refers to vehicle hours traveled. From these three figures, we can observe that the proposed methodology can predict the traffic flow with high accuracy, as the true VMT and VHT are close to the predicted VMT and VHT. For example, for the 5th-hour prediction task, the predicted VMT is 1.260 when the true VMT is 1.219. For the 1st-hour prediction task, the predicted VHT is 0.337 when the true VHT is 0.325. To further demonstrate the performance of the proposed method, we compare the proposed method with existing methods reported in the literature, and these methods are listed in Table 5. In this table, the D-ConvoLSTM method refers to the decoder network with the convolutional LSTM network; and the D-Attention method refers to the decoder network with the multi-head attention mechanism; LSTM refers to the long short-term memory network; LASSO refers to the least absolute shrinkage and selection operator; ANN refers to the artificial neural network.

Table 5. Symbols and descriptions of the proposed method and other methods for traffic flow predictions.

Method Symbol	Description
D-ConvLSTM	Decoder with convolutional LSTM
D-Attention	Decoder with multi-head attention
LSTM	Long short term memory network
LASSO	Regression with l1-norm regularization
ANN	Artificial neural network

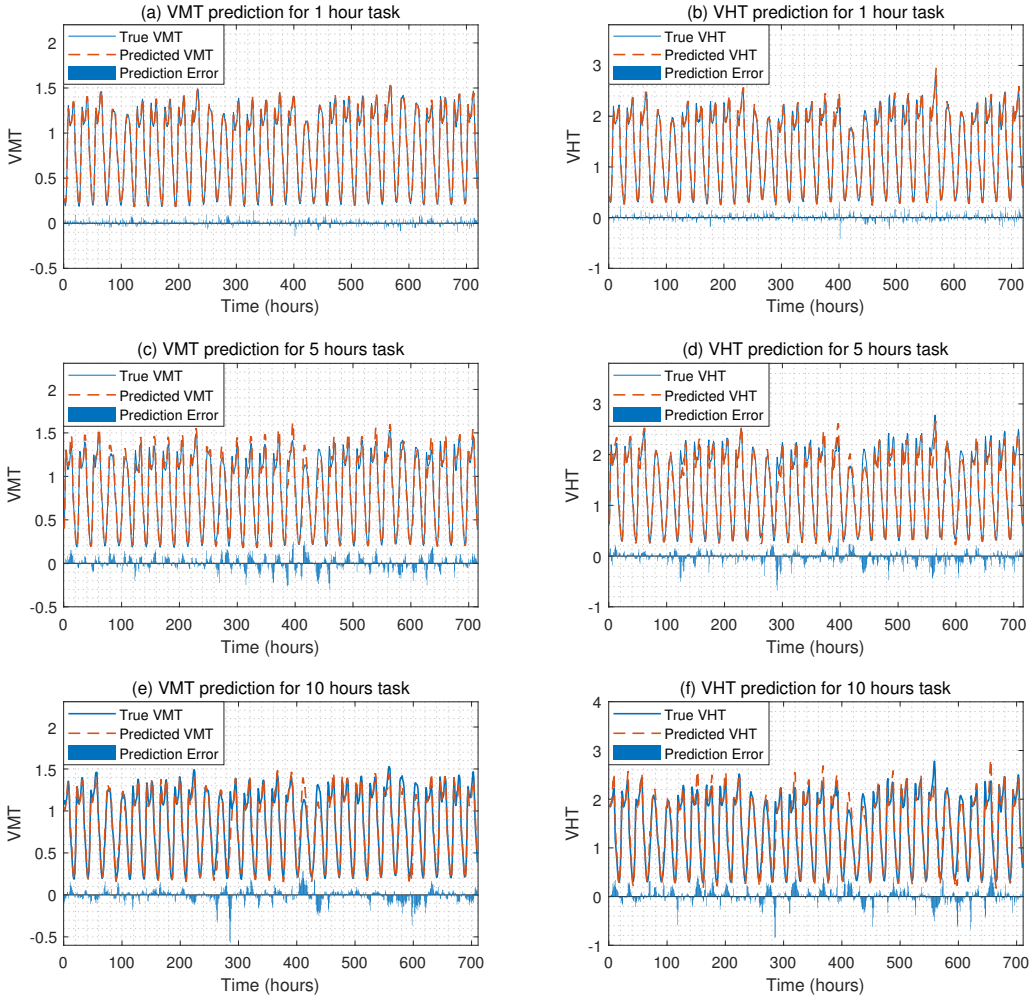


Figure 5. The VMT and VHT prediction results for three different tasks under the first case, where (**a**,**c**,**e**) show the VMT predictions for three tasks; and (**b**,**d**,**f**) show the VHT predictions for three tasks.

Table 6 compares the traffic flow prediction performance of the proposed method with methods listed in Table 5 in terms of RMSE and MAE. From this table, we can conclude that the proposed method can predict traffic flow with high accuracy and outperforms existing data-driven methods. For example, for the 1st-hour task, the RMSE of the VMT prediction for the proposed method is 0.032, and the RMSE of other data-driven methods ranges from 0.038 to 0.088. For the 5th-hour task, the RMSE of the VHT prediction for the proposed method is 0.128; however, the RMSE of LSTM is 0.145, and the RMSE of ANN is 0.245.

Table 6. The traffic flow prediction errors in terms of RMSE and MAE for the proposed methods and other data-driven methods under the first case.

		1 h Task		5 h Task		10 h Task	
		VMT	**VHT**	**VMT**	**VHT**	**VMT**	**VHT**
RMSE	Proposed	0.032	0.066	0.080	0.128	0.084	0.167
	D-ConvLSTM	0.044	0.079	0.099	0.128	0.094	0.157
	D-Attention	0.043	0.086	0.105	0.179	0.113	0.199
	LSTM [30]	0.038	0.064	0.065	0.145	0.104	0.191
	LASSO [48]	0.088	0.141	0.142	0.242	0.141	0.240
	ANN [49]	0.054	0.103	0.137	0.245	0.138	0.241
MAE	Proposed	0.024	0.048	0.059	0.090	0.058	0.116
	D-ConvLSTM	0.034	0.058	0.072	0.097	0.064	0.115
	D-Attention	0.034	0.066	0.076	0.135	0.077	0.138
	LSTM [30]	0.029	0.045	0.046	0.107	0.064	0.130
	LASSO [48]	0.063	0.096	0.099	0.165	0.098	0.163
	ANN [49]	0.039	0.072	0.090	0.172	0.089	0.168

4.5. Traffic Flow Prediction Results for the Second Case

Figure 6 shows the traffic flow prediction results for three different tasks under the second case, where VMT refers to vehicle miles traveled, and VHT refers to vehicle hours traveled. From this figure, we can observe that the proposed methodology can predict the traffic flow with high accuracy as the true VMT and VHT are close to the predicted VMT and VHT. For example, for the 5th-hour prediction task, the predicted VMT is 1.085 when the true VMT is 1.082. For the 1st-hour prediction task, the predicted VHT is 2.110 when the true VHT is 2.138. Table 7 compares the traffic flow prediction performance of the proposed method with methods listed in Table 5 in terms of RMSE and MAE. From this table, we can conclude that the proposed method can predict traffic flow with high accuracy and outperforms existing data-driven methods. For example, for the 1st-hour task, the RMSE of the VMT prediction for the proposed method is 0.053, and the RMSE of other data-driven methods ranges from 0.055 to 0.091. For the 5th-hour task, the MAE of the VHT prediction for the proposed method is 0.093; however, the RMSE of LSTM is 0.129, and the RMSE of ANN is 0.175.

Table 7. The traffic flow prediction errors in terms of RMSE and MAE for the proposed methods and other data-driven methods under the second case.

		1 h Task		5 h Task		10 h Task	
		VMT	**VHT**	**VMT**	**VHT**	**VMT**	**VHT**
RMSE	Proposed	0.053	0.100	0.084	0.135	0.100	0.172
	D-ConvLSTM	0.088	0.153	0.094	0.157	0.118	0.184
	D-Attention	0.055	0.087	0.112	0.168	0.141	0.253
	LSTM [30]	0.055	0.106	0.113	0.187	0.119	0.225
	LASSO [48]	0.091	0.145	0.149	0.256	0.145	0.248
	ANN [49]	0.063	0.112	0.143	0.255	0.146	0.256
MAE	Proposed	0.042	0.078	0.062	0.093	0.064	0.109
	D-ConvLSTM	0.060	0.107	0.060	0.104	0.078	0.125
	D-Attention	0.043	0.107	0.075	0.123	0.104	0.187
	LSTM [30]	0.046	0.076	0.077	0.129	0.080	0.153
	LASSO [48]	0.063	0.099	0.102	0.175	0.100	0.168
	ANN [49]	0.044	0.079	0.096	0.175	0.099	0.179

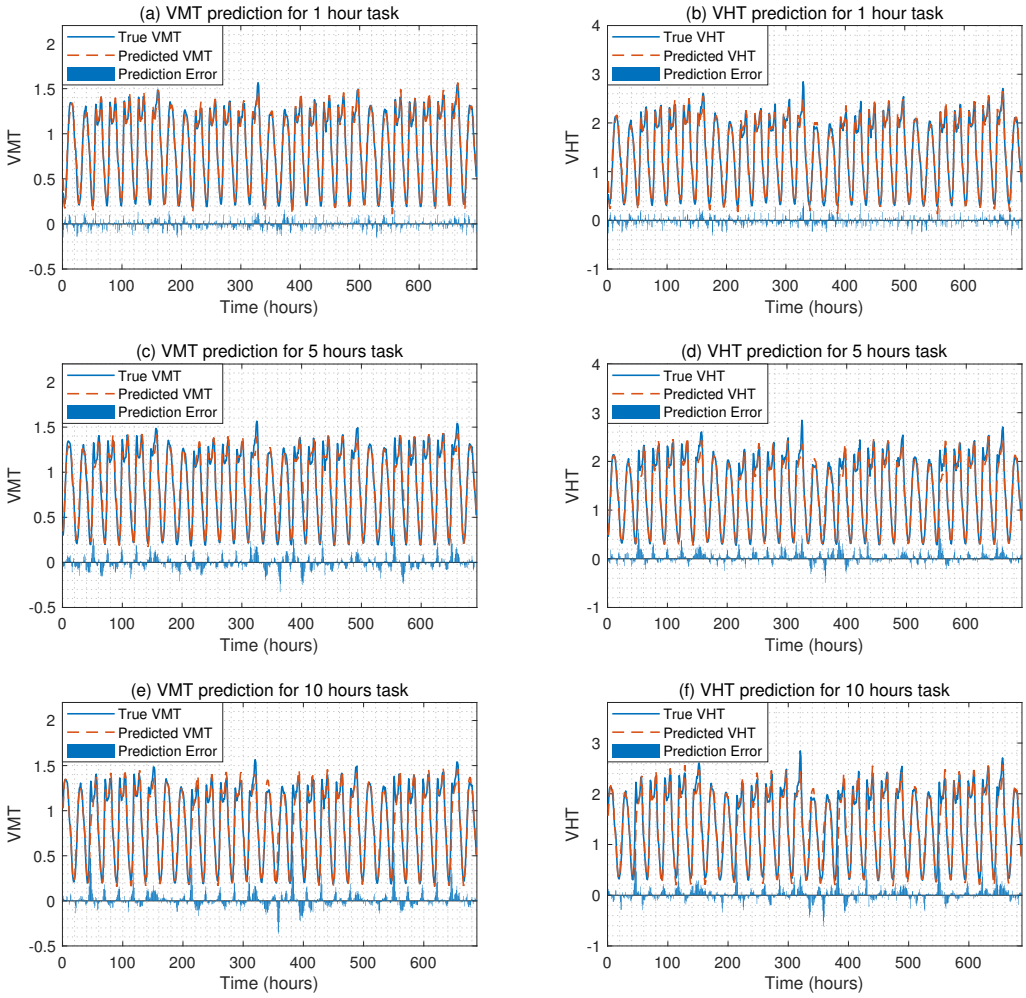


Figure 6. The VMT and VHT prediction results for three different tasks under the second case, where (**a**,**c**,**e**) show the VMT predictions for three tasks; and (**b**,**d**,**f**) show the VHT predictions for three tasks.

5. Conclusions and Future Work

In this study, a deep learning approach was proposed to predict traffic flow. In the proposed deep learning approach, a convolutional long short-term memory network was used to consider the correlation of the extracted high-dimensional features and the temporal correlation of traffic flow data in a unified manner. Moreover, a multi-head attention mechanism was implemented to use the most relevant portion of the traffic flow data to make predictions so that the prediction performance can be improved. A traffic flow dataset collected from the Caltrans Performance Measurement System (PeMS) database was used to demonstrate the effectiveness of the proposed method. Experimental results have shown that the proposed method can accurately predict the traffic flow with a minimum RMSE of 0.032 and outperforms the existing data-driven methods in terms of RMSE and MAE. Future work will be directed to use the convolutional LSTM network to make traffic flow predictions under more complicated environments and conditions.

Author Contributions: Y.W.: conceptualization, methodology, software, investigation, validation, visualization, funding acquisition, writing—original draft preparation. H.L.: conceptualization, writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Mineta Transportation Institute under Project No. 2211.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used in this study is available at https://pems.dot.ca.gov/ (accessed on 5 February 2022).

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

Abbreviations

The following abbreviations are used in this manuscript:

LSTM	Long short-term memory
ANN	artificial neural network
PeMS	Caltrans Performance Measurement System
VMT	vehicle miles traveled
VHT	vehicle hours traveled
RSME	root mean squared error
MAE	mean absolute error
FC	fully connected
ConvLSTM	convolutional long short-term memory

References

1. Bazzan, A.L.; Oliveira, D.d.; Klügl, F.; Nagel, K. To adapt or not to adapt–consequences of adapting driver and traffic light agents. In *Adaptive Agents and Multi-Agent Systems III. Adaptation and Multi-Agent Learning*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 1–14.
2. Ahmad, A.; Arshad, R.; Mahmud, S.A.; Khan, G.M.; Al-Raweshidy, H.S. Earliest-deadline-based scheduling to reduce urban traffic congestion. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 1510–1526. [CrossRef]
3. Zhang, Y.; Li, C.; Luan, T.H.; Fu, Y.; Shi, W.; Zhu, L. A mobility-aware vehicular caching scheme in content centric networks: Model and optimization. *IEEE Trans. Veh. Technol.* **2019**, *68*, 3100–3112. [CrossRef]
4. Falcocchio, J.C.; Levinson, H.S. *Road Traffic Congestion: A Concise Guide*; Springer: Berlin/Heidelberg, Germany, 2015; Volume 7.
5. Wu, Y.; Tan, H.; Qin, L.; Ran, B.; Jiang, Z. A hybrid deep learning based traffic flow prediction method and its understanding. *Transp. Res. Part C Emerg. Technol.* **2018**, *90*, 166–180. [CrossRef]
6. Shi, R.; Du, L. Multi-Section Traffic Flow Prediction Based on MLR-LSTM Neural Network. *Sensors* **2022**, *22*, 7517. [CrossRef] [PubMed]
7. Wang, S.; Zhao, J.; Shao, C.; Dong, C.; Yin, C. Truck traffic flow prediction based on LSTM and GRU methods with sampled GPS data. *IEEE Access* **2020**, *8*, 208158–208169. [CrossRef]
8. Chen, Z.; Wu, B.; Li, B.; Ruan, H. Expressway exit traffic flow prediction for ETC and MTC charging system based on entry traffic flows and LSTM model. *IEEE Access* **2021**, *9*, 54613–54624. [CrossRef]
9. Zhou, Q.; Chen, N.; Lin, S. FASTNN: A Deep Learning Approach for Traffic Flow Prediction Considering Spatiotemporal Features. *Sensors* **2022**, *22*, 6921. [CrossRef]
10. Yu, C.; Chen, J.; Xia, G. Coordinated Control of Intelligent Fuzzy Traffic Signal Based on Edge Computing Distribution. *Sensors* **2022**, *22*, 5953. [CrossRef]
11. Feng, X.; Ling, X.; Zheng, H.; Chen, Z.; Xu, Y. Adaptive multi-kernel SVM with spatial–temporal correlation for short-term traffic flow prediction. *IEEE Trans. Intell. Transp. Syst.* **2018**, *20*, 2001–2013. [CrossRef]
12. Kumar, S.V. Traffic flow prediction using Kalman filtering technique. *Procedia Eng.* **2017**, *187*, 582–587. [CrossRef]
13. Mingheng, Z.; Yaobao, Z.; Ganglong, H.; Gang, C. Accurate multisteps traffic flow prediction based on SVM. *Math. Probl. Eng.* **2013**, *2013*, 418303. [CrossRef]
14. 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]
15. Miglani, A.; Kumar, N. Deep learning models for traffic flow prediction in autonomous vehicles: A review, solutions, and challenges. *Veh. Commun.* **2019**, *20*, 100184. [CrossRef]
16. Sun, Y.; Leng, B.; Guan, W. A novel wavelet-SVM short-time passenger flow prediction in Beijing subway system. *Neurocomputing* **2015**, *166*, 109–121. [CrossRef]
17. Liu, Y.; Wu, H. Prediction of road traffic congestion based on random forest. In Proceedings of the 2017 10th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 9–10 December 2017; IEEE: Piscataway, NJ, USA, 2017; Volume 2, pp. 361–364.
18. Sun, S.; Xu, X. Variational inference for infinite mixtures of Gaussian processes with applications to traffic flow prediction. *IEEE Trans. Intell. Transp. Syst.* **2010**, *12*, 466–475. [CrossRef]
19. Pascale, A.; Nicoli, M. Adaptive Bayesian network for traffic flow prediction. In Proceedings of the 2011 IEEE Statistical Signal Processing Workshop (SSP), Nice, France, 28–30 June 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 177–180.

20. Tang, J.; Chen, X.; Hu, Z.; Zong, F.; Han, C.; Li, L. Traffic flow prediction based on combination of support vector machine and data denoising schemes. *Phys. Stat. Mech. Its Appl.* **2019**, *534*, 120642. [CrossRef]
21. Zhang, L.; Alharbe, N.R.; Luo, G.; Yao, Z.; Li, Y. A hybrid forecasting framework based on support vector regression with a modified genetic algorithm and a random forest for traffic flow prediction. *Tsinghua Sci. Technol.* **2018**, *23*, 479–492. [CrossRef]
22. Xu, Y.; Yin, F.; Xu, W.; Lin, J.; Cui, S. Wireless traffic prediction with scalable Gaussian process: Framework, algorithms, and verification. *IEEE J. Sel. Areas Commun.* **2019**, *37*, 1291–1306. [CrossRef]
23. Wang, W.; Zhou, C.; He, H.; Wu, W.; Zhuang, W.; Shen, X. Cellular traffic load prediction with LSTM and Gaussian process regression. In Proceedings of the ICC 2020-2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–6.
24. Zhu, Z.; Peng, B.; Xiong, C.; Zhang, L. Short-term traffic flow prediction with linear conditional Gaussian Bayesian network. *J. Adv. Transp.* **2016**, *50*, 1111–1123. [CrossRef]
25. Li, Z.; Jiang, S.; Li, L.; Li, Y. Building sparse models for traffic flow prediction: An empirical comparison between statistical heuristics and geometric heuristics for Bayesian network approaches. *Transp. Transp. Dyn.* **2017**, *7*, 107–123. [CrossRef]
26. Wei, W.; Wu, H.; Ma, H. An autoencoder and LSTM-based traffic flow prediction method. *Sensors* **2019**, *19*, 2946. [CrossRef]
27. Xiao, Y.; Yin, Y. Hybrid LSTM neural network for short-term traffic flow prediction. *Information* **2019**, *10*, 105. [CrossRef]
28. 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; IEEE: Piscataway, NJ, USA, 2016; pp. 324–328.
29. Shu, W.; Cai, K.; Xiong, N.N. A short-term traffic flow prediction model based on an improved gate recurrent unit neural network. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 16654–16665. [CrossRef]
30. Yang, B.; Sun, S.; Li, J.; Lin, X.; Tian, Y. Traffic flow prediction using LSTM with feature enhancement. *Neurocomputing* **2019**, *332*, 320–327. [CrossRef]
31. Xiangxue, W.; Lunhui, X.; Kaixun, C. Data-driven short-term forecasting for urban road network traffic based on data processing and LSTM-RNN. *Arab. J. Sci. Eng.* **2019**, *44*, 3043–3060. [CrossRef]
32. Li, Z.; Xiong, G.; Chen, Y.; Lv, Y.; Hu, B.; Zhu, F.; Wang, F.Y. A hybrid deep learning approach with GCN and LSTM for traffic flow prediction. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealan, 27–30 October 2019; IEEE: Piscatway, NJ, USA, 2019; pp. 1929–1933.
33. Chen, J.; Liao, S.; Hou, J.; Wang, K.; Wen, J. GST-GCN: A Geographic-Semantic-Temporal Graph Convolutional Network for Context-aware Traffic Flow Prediction on Graph Sequences. In Proceedings of the 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Melbourne, Australia, 17–20 October 2021; IEEE: Piscataway, NJ, USA, 2020; pp. 1604–1609.
34. Jiang, W.; Luo, J. Graph neural network for traffic forecasting: A survey. *Expert Syst. Appl.* **2022**, *4*, 117921. [CrossRef]
35. Tian, Y.; Zhang, K.; Li, J.; Lin, X.; Yang, B. LSTM-based traffic flow prediction with missing data. *Neurocomputing* **2018**, *318*, 297–305. [CrossRef]
36. Dai, G.; Ma, C.; Xu, X. Short-term traffic flow prediction method for urban road sections based on space–time analysis and GRU. *IEEE Access* **2019**, *7*, 143025–143035. [CrossRef]
37. Zhene, Z.; Hao, P.; Lin, L.; Guixi, X.; Du, B.; Bhuiyan, M.Z.A.; Long, Y.; Li, D. Deep convolutional mesh RNN for urban traffic passenger flows prediction. In Proceedings of the 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), Guangzhou, China, 8–12 October 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1305–1310.
38. Luo, X.; Li, D.; Yang, Y.; Zhang, S. Spatiotemporal traffic flow prediction with KNN and LSTM. *J. Adv. Transp.* **2019**, *2019*, 4145353. [CrossRef]
39. Zhu, H.; Xie, Y.; He, W.; Sun, C.; Zhu, K.; Zhou, G.; Ma, N. A novel traffic flow forecasting method based on RNN-GCN and BRB. *J. Adv. Transp.* **2020**, *2020*, 7586154. [CrossRef]
40. Yu, B.; Lee, Y.; Sohn, K. Forecasting road traffic speeds by considering area-wide spatio-temporal dependencies based on a graph convolutional neural network (GCN). *Transp. Res. Part C Emerg. Technol.* **2020**, *114*, 189–204. [CrossRef]
41. Ye, J.; Zhao, J.; Ye, K.; Xu, C. How to build a graph-based deep learning architecture in traffic domain: A survey. *IEEE Trans. Intell. Transp. Syst.* **2020**, *2020*, 7586154. [CrossRef]
42. 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**, *23*, 3904–3924.
43. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv* **2014**, arXiv:1409.0473.
44. Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In Proceedings of the International Conference on Machine Learning, PMLR, Lille, France, 7–9 July 2015; pp. 2048–2057.
45. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems 30, Long Beach, CA, USA, 4–9 December 2017; Volume 30.
46. Li, J.; Tu, Z.; Yang, B.; Lyu, M.R.; Zhang, T. Multi-head attention with disagreement regularization. *arXiv* **2018**, arXiv:1810.10183.
47. Caltrans. Performance Measurement System (PeMS). Available online: https://pems.dot.ca.gov/ (accessed on 5 February 2022).

48. Sun, S.; Huang, R.; Gao, Y. Network-scale traffic modeling and forecasting with graphical lasso and neural networks. *J. Transp. Eng.* **2012**, *138*, 1358–1367. [CrossRef]
49. Rahman, F.I. Short term traffic flow prediction using machine learning-KNN, SVM and ANN with weather information. *Int. J. Traffic Transp. Eng.* **2020**, *10*, 371–389.