Article

Intercity Online Car-Hailing Travel Demand Prediction via a Spatiotemporal Transformer Method

Hongbo Li ¹, Jincheng Wang ², Yilong Ren ^{1,3,4,*} and Feng Mao ⁵

¹ School of Transportation Science and Engineering, Beihang University, Beijing 100191, China; lihongbo@buaa.edu.cn

² School of Economics and Management, Chang'an University, Xi'an 710064, China; wangjincheng@chd.edu.cn

³ Beihang Hangzhou Innovation Institute Yuhang, Hangzhou 310023, China

⁴ National Engineering Laboratory for Comprehensive Transportation Big Data Application Technology, Beijing 100191, China

⁵ Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen 518055, China; mfl19@mails.tsinghua.edu.cn

* Correspondence: yilongren@buaa.edu.cn

Abstract: Traffic prediction is a critical aspect of many real-world scenarios that requires accurate traffic status predictions, such as travel demand prediction. The emergence of online car-hailing activities has given people greater mobility and makes intercity travel more frequent. The increase in online car-hailing demand has often led to a supply–demand imbalance where there is a mismatch between the immediate availability of car-hailing services and the number of passengers in certain areas. Accurate prediction of online car-hailing demand promotes efficiencies and minimizes resources and time waste. However, many prior related studies often fail to fully utilize spatiotemporal characteristics. With the development of newer deep-learning models, this paper aims to solve online car-hailing problems with an ST-transformer model. The spatiotemporal characteristics of online car-hailing data are analyzed and extracted. The study region is divided into subareas, and the demand for each subarea is summed at a specific time interval. Historical demand of the areas is used to predict future demand. The results of the ST-transformer outperformed other baseline models, namely, VAR, SVR, LSTM, LSTNet, and transformers. The validated results suggest that the ST-transformer is more capable of capturing spatiotemporal characteristics compared to the other models. Additionally, compared to others, the model is less affected by data sparsity.

Keywords: intercity car-hailing; travel demand; prediction; spatiotemporal transformer

Citation: Li, H.; Wang, J.; Ren, Y.; Mao, F. Intercity Online Car-Hailing Travel Demand Prediction via a Spatiotemporal Transformer Method. *Appl. Sci.* **2021**, *11*, 11750. <https://doi.org/10.3390/app112411750>

Academic Editor: Nadia Giuffrida

Received: 25 November 2021

Accepted: 8 December 2021

Published: 10 December 2021

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

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

1. Introduction

China has developed numerous urban agglomerations as the economy and transportation networks have progressed. Some examples include the Yangtze River Delta economic circle, the Pearl River Delta economic circle, the Beijing–surroundings economic circle, the Sichuan–Chongqing economic circle, and the Yinchuan metropolitan area. Intercity travel activities, such as commuting across cities, and travel between residential and urban transport facilities, such as railway stations and airports, have become more popular as a result. With the rapid growth of transit infrastructures, services such as online intercity car-hailing have gained popularity in recent years. The convenience of online booking and on-demand services of online intercity car-hailing has been well received by users. As reported by China's Ministry of Transportation, 660 million car-hailing orders were placed in November 2020 alone, and most of them were related to intercity travel.

However, such an increase in online intercity car-hailing demand has often led to a supply–demand imbalance where there is a mismatch between the immediate availability of car-hailing services and the number of passengers in certain areas. To maintain a more well-balanced spatial distribution of vacant taxis to meet the demand in certain areas

and thereby enhance the efficiency of online car-hailing services, demand forecasting for intercity online car-hailing travel is vital. Therefore, this issue can be solved as a travel demand forecasting problem.

Extensive studies have been conducted to increase the precision of travel demand prediction. Existing solutions to travel demand prediction problems can be classified into two major categories: statistical methods and deep-learning methods. Initially, most of the forecasting is conducted using statistical models such as autoregressive integrated moving average (ARIMA) and other alternatives. Li et al. explored human mobility patterns by using an ARIMA-based method [1]. Other studies [2–4] further considered factors such as spatial relationships and weather conditions with remarkable successes. However, traffic data have inherent spatiotemporal dependencies, which makes traffic prediction a highly challenging and complex task. ARIMA and other statistical methods still fail to manage such complex nonlinear relationships. Furthermore, optimizations of these models and overfitting remain very large challenges to address [5].

To address the above challenges, more recent attempts have focused on deep-learning-based methods, as these methods can well fit nonlinear relationships between future travel and historical travel demand. Deep-learning-based methods have gained remarkable performance in many learning tasks [6], which has inspired applications of deep-learning techniques for traffic prediction problems. Commonly used deep-learning models for traffic prediction tasks can be classified into three categories, namely, feedforward neural networks, recurrent neural networks (RNNs), and convolutional neural networks (CNNs) [7].

A multilayer perceptron (MLP) is an example of a simple feedforward neural network, as exemplified by the work done by Wong et al. in 2017 [8]. An MLP was used to model both the supply and demand of passenger and taxi services. However, these models fail to consider both spatial and temporal interactions. Recurrent neural networks (RNNs) can use self-circulation mechanisms and deal with temporal dependencies effectively [5,7]. Hence, RNN models and their variants are also applied by many researchers. The long short-term memory (LSTM) model developed by Hochreiter and Schmidhuber in 1997 [9] is one of the RNN variants. It is a gated recurrent neural network that includes an extra hidden state compared to the original RNN. This enables LSTM models to be more efficient at capturing longer time dependencies and resolves the problem of vanishing and/or exploding gradient problems in the original RNN to a certain extent. This characteristic makes LSTM models a popular option in short-term traffic forecasting problems [10–12]. Yu et al. [13] employed an LSTM network to predict traffic under extreme conditions, proving LSTM's capability in modeling sequential dependency. Yao et al. [14] used LSTM that considered semantic similarity among different regions to make predictions. However, these sequential models have limited scalability for longer time dependencies where their memorization power declines [7]. Furthermore, sequential models are good at temporal dependencies but have no mechanism to deal with the spatial dependencies in traffic data. In practice, LSTM is often combined with models that capture spatial features, such as convolutional neural networks (CNNs), to form hybrid models. Such a hybrid model is then able to capture the temporal–spatial characteristics of traffic data.

CNNs have shown outstanding performance in modeling local and shift-variant features [15]. Lv et al. [16] further integrated an RNN and CNN, whereby the latter was in charge of spatial features and used LSTM to capture temporal features. The long- and short-term temporal network (LSTNet), proposed by Lau et al. [17], is another example of LSTM and CNN integration that demonstrated significant performance improvements for long- and short-time series prediction tasks. However, since the number of hidden layers grows linearly with increasing sequences, the scalability of such models is limited for longer input sequences [18]. Deeper layers also affect the efficiency of CNNs for capturing dependencies in longer sequences [19].

After a review of related works, the utilization of spatiotemporal features is key to forecasting travel demand with high accuracy. However, all the above research mainly focused on intracity travel demand forecasting, and few studies have focused on the intercity travel

demand prediction problem. In addition, compared to intracity travel demand prediction, two challenges need to be solved for intercity travel demand forecasting. First, it requires capturing a long time-horizon correlation. Second, the correlation among adjacent regions may not be obvious, and many regions have high correlations with some far away regions.

To conduct accurate intercity travel demand prediction, we first utilized a transformer model to capture longer time-horizon correlations. A transformer is a popular model proposed by Vaswani et al. [19]. It is an attention-based model that no longer requires recursive feeding of sequential data, as RNN-based models do. Therefore, a transformer can preserve sequence orders and is more computationally efficient than RNNs. Strategies such as multiheaded attention and positional encoding help a transformer attain significant success in machine translation tasks [20,21]. Machine translation and traffic prediction are formulated similarly. Machine translation is a seq-2-seq learning task that aims to translate a source sentence into a target sentence [22], whereas traffic prediction tries to use historical data as an indicator of future traffic conditions. Hence, the time steps in historical traffic data can be similar to the position index of each word in the input sentences in a machine translation task [23]. For instance, Cai et al. [23] utilized a transformer to capture temporal dependencies in traffic data.

To tackle the second challenge, a spatial transformer was introduced that armed the transformer with spatial transformation capabilities [24]. If we employ a CNN model for spatial dependency extraction, the representation of correlations between two distanced regions usually requires multiple hidden layers. The spatial transformer can capture the correlation between two regions and is independent of the distance between regions.

To conclude, we will address online car-hailing demand prediction tasks using a state-of-art model spatial-temporal transformer (ST-transformer). We hope that the spatial-temporal modeling capability of the ST-transformer can make accurate traffic predictions about travel demands and thereby provide suggestions for a more well-balanced spatial distribution of vacant taxis to meet the demand in specific areas and thereby enhance the efficiency of online car-hailing services and reduce resource waste. Most of the prior works divide the studied region into smaller subareas and tabulate travel demands in each subarea during a time interval [14,25–27]; hence, similar methods will be adopted by this paper. We will test the capacity of other models, such as LSTM, LSTNet, and a transformer, and compare them with the ST-transformer.

The rest of this paper is organized as follows. Section 2 outlines the study area and dataset used, including data processing and extraction methodologies. Section 3 defines the problem in detail and lists the methodologies adopted, with elaborations on model architectures. Section 4 describes our experimental design and presents the performances of the various models. This paper ends with conclusions and future ideas in Section 5.

2. Data Description

2.1. Study Area

Our online car-hailing data are distributed in Yinchuan City and Shizuishan City, Ningxia Autonomous Region, China, and the geographical position is shown in Figure 1. Yinchuan City, located at 105.82–106.88 E longitude, 37.59–38.88 N latitude, is the capital of Ningxia, serving as one of the most important cities in Northwest China with more than two million people. Shizuishan City, located at 105.96–106.97 E longitude, 38.60–39.39 N latitude, is north of Yinchuan City with a resident population of approximately 800,000, serving as an important pillar of Yinchuan.

To obtain spatiotemporal data, we classify the orders into 30,000 grids with a size of 1240×1530 m. Finally, 357 grids are selected with a record of at least one order.

2.2. Intercity Car-Hailing Data

There are 224,822 orders in our dataset, ranging from 1 January to 31 December 2020, and all orders are located within the area shown in Figure 1.

Figure 1. The study area. (a) Orders in the study area. (b) Selected grids of the study area.

Each order records the spatial information, temporal information, passenger information, and so on. These are typical spatiotemporal data that include the time when the passengers order online and get into the car, the location where the passengers get into and out of the car, and some desensitized passenger data such as number and age, which are listed and described in Table 1.

Table 1. Intercity car-hailing data fields and descriptions.

Fields	Description
Order time	The time when the passengers order online.
Departure time	The time when the passengers get into the car.
Departure location	Latitude and longitude of departure.
Destination location	Latitude and longitude of destination.
Number of passengers	The number of passengers getting into the car.
Ages of passengers	Age of each passenger of the order.

2.3. Data Processing

The data coverage area of this paper is consistent with the study area. Note that data in February are seriously distorted due to COVID-19; therefore, we include only data from March 1st to December 31st for temporal coherence, and we highlight the temporal attribute. As mentioned in Section 2.1, only 357 grid units with at least one order are selected. In addition, the dataset is split by chronological order into a training set, validation set, and testing set at a ratio of 7:2:1, and Z-score normalization is applied to the inputs.

Two prediction scenarios are designed to compare the performance of the ST-transformer and other baselines. The passenger volume in the previous 6 time steps is used to predict the volume in the next time step.

Scenario 1: 1 h traffic demand prediction using the last 6 h order distribution;

Scenario 2: 2 h traffic demand prediction using the last 12 h order distribution.

Using the two scenarios above, the performance of the ST-transformer in the long- and short-term is compared with the other models.

3. Methodology

3.1. Problem Definition

The input X is a total matrix, recording the passenger flow in different places at different times. Matrix element x_t^s is the volume of passenger flow in grid s at time t . Whereby s varies from 1 to 357, representing the 357 observation points, which correspond

to the passenger flow in the 357 grids. The numerical value of t varies within the limit of one year.

$$\mathbf{X} = \begin{pmatrix} x_1^1 & \cdots & x_1^s \\ \vdots & \ddots & \vdots \\ x_t^1 & \cdots & x_t^s \end{pmatrix} \quad (1)$$

The output Y is the passenger flow in the next time step, $t + 1$ as follows:

$$\mathbf{Y} = [y_{t+1}^1, y_{t+1}^2, \dots, y_{t+1}^s] \quad (2)$$

3.2. Spatiotemporal Transformer Net

A spatiotemporal transformer net (ST-transformer) was used to predict the traffic flow. Noting the unevenness of the order distribution, we believed that an attention layer was what we needed for effective trajectory prediction.

The ST-transformer intends to interleave the spatial and temporal transformer networks in a single frame to solve the combined prediction problem with spatiotemporal data. And the model of a ST-transformer network is shown in Figure 2. The spatial transformer focuses on the topological structure of the data and calculates the connection value between each node. The temporal transformer is a simple part of the transformer network that pays attention to sequence continuity.

Figure 2. Model of a spatiotemporal transformer network.

Both the spatial part and temporal part end with a specifically designed activation function called position-wise feed-forward (PFF), as shown in Figure 3. PFF can be calculated as:

$$f(x) = \max(0, xW_1 + b_1)W_2 + b_2 \quad (3)$$

The start is a convolution layer with 8 kernels of size (1, 1), which aims to amplify the data from (20, 357, 2, 6) to (20, 357, 8, 6). The input matrix can be written as X_{input} , or X_i after the start convolution layer. The encoder is composed of a spatial transformer (ST) net and a temporal transformer (TT) net:

$$\mathbf{X}_e^S = f(ST_e(\mathbf{X}_i)) \quad (4)$$

$$\mathbf{X}_e^T = f(TT_e(\mathbf{X}_i)) \quad (5)$$

$$\mathbf{X}_e = \mathbf{X}_e^S \oplus \mathbf{X}_e^T \quad (6)$$

where $\oplus$ identifies a concatenation operation that doubles the dimension of the channels to $(20, 357, 6, 2 \times 8)$; X_i denotes the output from the start convolution part; and X_e^T and X_e^S denote the results of ST and TT for the encoder, respectively.

Figure 3. Structure of the PFF activation function.

After two layers of convolution to reduce the hidden variables, we obtain a feature matrix X_F of size $(20, 1, 357, 8)$. The TT and ST parts are set in parallel and will be discussed in Sections 3.3 and 3.4.

The decoder adapts a series method to connect the ST and TT:

$$X_d^T = f(TT_e(X_F)) \quad (7)$$

$$X_d^S = f(ST_d(X_d^T)) \quad (8)$$

$$X_d = X_d^S \quad (9)$$

In the post-convolution stage, 64 kernels with a size of $(1, 1)$ and ReLU activation are used to increase the dimension. Finally, we utilize a convolution layer to obtain the predicted matrix X_{output} of size $(20, 357, 1, 2)$.

3.3. Temporal Transformer

By using transformers, as efficient deep-learning models based on a self-attention mechanism, great achievements have been witnessed in many fields, such as natural language processing (NLP), computer vision (CV), and deep learning for graphs. In this article, the transformer networks are all designed to capture spatial and temporal features.

The TT part uses a multi-head attention mechanism found in transformer networks. Figure 4 shows the calculation process for a single attention mechanism, and here, a matrix calculation method is given. This self-attention mechanism aims to calculate the correlations among vectors in the input matrix X , which is often calculated by softmax. To improve the fitting ability of this model, the weight matrix is added to define query matrix $Q = XW^Q$, key matrix $K = XW^K$, and value matrix $V = XW^V$. The self-attention mechanism can be concluded in four steps: firstly, the similarity is calculated by QK^T ; secondly, the result is divided by $\sqrt{d_k}$ for normalization where d_k means the dimension of K ; thirdly, the weighted sum of V is calculated with the weight matrix above; finally, softmax is used for a probability. The attention formula is as follows.

$$ATT(Q, K, V) = \frac{\text{Softmax}(QK^T)}{\sqrt{d_k}} V \quad (10)$$

A mask with the shape of a strictly upper triangular matrix is used to reduce label leakage. Before the fully connected multi-head layer, 30% of the neurons are designed to die to restrain overfitting.

Figure 4. Model of the temporal transformer.

A residual mechanism block is added to the TT, and the input of each layer is the sum of the output and input of the previous layer, which solves the vanishing gradient problem. The head formula is:

$$head_k = ATT(QW_k^Q, KW_k^K, VW_k^V) \quad (11)$$

where k varies from 1 to 8, and the multi-head mechanism outputs a tensor composed of the k heads:

$$MultiHead(Q, K, V) = (head_1 \oplus \dots \oplus head_k)W^O \quad (12)$$

where all W ($W_k^Q, W_k^V, W_k^K, W^O, etc$) indicate a weight matrix generated by a fully connected layer. (Q, K, V) is a function with the independent variables of $\{X_i^t\}_{t=1}^T$. So, the overall formula of this mechanism is:

$$X_O = ST(X_i) = X_i + MultiHead(X_i) \quad (13)$$

3.4. Spatial Transformer

From the equations in 3.3, it is noted that the transformer network pays little attention to the structural information of the input; consequently, we utilize the ST block to extract the spatial feature of the trajectory and mine the correlations between grids on the map.

As shown in Figure 5, the geographical distribution can be treated as a graph $G = (V, E)$, where $V = \{1, 2, \dots, n\}$ is the grid set and $E = \{(i, j) | i, j \text{ is connected}\}$ is the edge set. This modeling method is also known as transformer-based graph convolution (TGConv). The graph varies over time and can be described as $g = \{G_1, \dots, G_t\}$, where t denotes the whole period.

Figure 5. Graph traffic model.

Assume that node i is connected with an embedding vector h_i and a neighbor set $Nb(i)$, where h_i is the feature vector of the feature set $\{h_t\}_{t=1}^T$. Defining the trajectory from

grid i to grid j in this fully connected graph as $m^{i \rightarrow j} = q_i^T k_j$, the attention mechanism in Formula (10) is then rewritten as:

$$Att(i) = \frac{Softmax([m^{i \rightarrow j}]_{j \in Nb(i) \cup \{i\}})}{\sqrt{d_k}} [v_j]_{j \in Nb(i) \cup \{i\}}^T + h_i \quad (14)$$

$$h'_i = Att(i) + f_{out}(Att(i)) \quad (15)$$

where output function f_{out} is designed as a fully connected layer, and h'_i is the updated embedding of i using the TGConv. The idea of ResNet and layer normalization are also applied in the ST.

With well-designed graph vertices and intersections, the ST works as well as the TT in the multi-head framework.

4. Results

This section presents the performance evaluation of the ST-transformer using a real-world dataset. We will use the online car-hailing dataset of recorded trips between Yinchuan and Shizuishan, China. The study areas are divided into 357 grids, and the passenger flow in each grid is aggregated into 1 h and 2 h windows.

4.1. Evaluation Metrics

The performance of the ST-transformer is evaluated in terms of the mean absolute error (MAE):

$$MAE = \frac{1}{N} \sum_{i=1}^N |\hat{y}_i - y_i| \quad (16)$$

mean absolute percentage error (MAPE):

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{\hat{y}_i - y_i}{y_i} \right| \quad (17)$$

and root mean squared error (RMSE):

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2} \quad (18)$$

The MAE represents the average of the absolute differences between the predicted passenger volume and actual passenger volume. This metric reflects the actual predicted value error. The MAPE is the percentage difference between the error and the actual passenger volume. Finally, the standard deviation of the error between the predicted value and the actual value is represented by the RMSE. A good prediction is indicated by lower values of all three metrics. Due to the presence of zero-valued grids, the scores of such grids are excluded for all evaluation metrics.

4.2. Baselines

To validate the effectiveness of the ST-transformer, it is compared with classic statistical models and deep-learning models. State-of-art models such as LSTNet and a transformer are also included in the comparison:

- VAR: The vectorized autoregressive model [28] is a generalized autoregressive model that captures the relations of multiple variables over time.
- SVR: Support vector regression [29] is a nonlinear regression model.
- LSTM: Long short-term memory [11] is a variant of an RNN that is more efficient at capturing longer time dependencies than an original RNN.

- LSTNet: A long- and short-term temporal network [17] is an integration of LSTM and a CNN that demonstrated significant performance improvements for long- and short-time series prediction tasks.
- A transformer [19] is naturally more computationally efficient than RNNs and can capture temporal features with an attention mechanism.

4.3. Experimental Results

Table 2 compares the performance of the ST-transformer and baseline models for the 1 h and 2 h window data. Prediction of the passenger volume for the next time steps uses the previous six time step data from the online car-hailing dataset recording trips between Yinchuan and Shizuishan.

Table 2. MAE, MAPE, and RMSE performances of the ST-transformer and baseline models.

Model	Performance					
	1 h Window			2 h Window		
	MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE
VAR	1.41	77.45%	2.01	1.71	73.25%	2.79
SVR	1.05	88.98%	1.16	1.15	89.76%	1.29
LSTM	1.08	38.74%	1.36	1.29	40.20%	2.05
LSTNet	1.07	65.32%	1.50	1.22	59.61%	1.95
Transformer	0.72	29.51%	1.50	1.07	35.97%	2.31
ST-Transformer	0.98	21.87%	1.06	0.76	51.51%	1.12

The ST-transformer obtains an outstanding result for both time windows compared to the other neural network-based and statistical-based methods. Generally, all the neural network-based models outperformed the VAR and SVR statistical-based methods. Although the SVR exhibits somewhat good MAE results for both time windows, the MAPE is extremely high, suggesting the possibility that incorrectly predicted values deviate greatly from the actual values. The transformer uses an attention mechanism to capture temporal features and outperforms LSTM and LSTNet. Both networks have inherent weaknesses in capturing longer time series. When passenger volume is aggregated based on a 2 h window, the performance of all models generally drops as the data become sparser. In contrast, the ST-transformer still excels and shows better performance than for the data aggregated based on a 1 h window. This may imply that the ST-transformer is less affected by the sparsity of the data, which is attributed to it being able to utilize spatial characteristics better.

As shown in Figure 6, the traffic demands are distributed extremely unevenly, and the orders are concentrated in several grids, which makes it difficult for traditional prediction methods. LSTM, VAR, and other models tend to forecast the orders in all grids below ten, as most grids have few car-hailing demands. Therefore, a large error is generated in specific grids with larger demands determined by geographical characteristics (traffic center, commercial building, residential quarters, and so on).

This characteristic is noticed by the attention mechanism of the ST-transformer, which performs desirably in Grid 50 to Grid 100. For some extreme cases, the ST-transformer still behaves better than other methods by nearly 25%.

Figure 7 shows the temporal error distribution of the various models. The traffic demands displayed periodic patterns for different days in a month, suggesting a possible trend in weekend and weekday distributions. Supporting previous observations, the ST-transformer obtained a rather stabilized performance in all three metrics, with a relatively low MAPE compared to all other models. We can conclude from this observation that the ST-transformer is indeed predicting with a smaller deviation from the ground truth.

Figure 6. Spatial error distribution for the different models. (a) MAE error. (b) MSE error.

Figure 7. Temporal error distribution for different models. (a) MAE. (b) MSE. (c) RMSE.

5. Conclusions

To enhance the efficiency of online car-hailing services, a spatial-temporal transformer model is used to make accurate travel demand predictions. A multi-head transformer attention mechanism is designed to reflect the temporal correlations. The graph model is established based on the geographical region for the calculation formula of the self-attention mechanism. The results showed that the ST-transformer produces less prediction error than LSTM, LSTN, transformer, and many other classical models. The data demonstrated that the advantages are more confident when predicting the near future. The transformer

mechanism based on the graph shows great superiority to deal with dynamical graph feature learning.

More accurate demand forecasting can help deploy more online car-hailing in places with high demand, help evacuate people, and increase road occupancy rates to reduce traffic congestion. Furthermore, improving the efficiency of online car-hailing services can increase people's confidence toward using online car-hailing to travel, thereby reducing the use of private cars on the road. Therefore, we believe that a large number of hired vehicles will reduce the chance of congestion on the roads.

Author Contributions: Conceptualization, H.L.; methodology, F.M.; software, J.W.; validation, Y.R.; formal analysis, H.L.; resources, J.W.; data curation, F.M.; writing—original draft preparation, Y.R.; writing—review and editing, J.W.; visualization, F.M.; supervision, J.W.; project administration, H.L.; funding acquisition, Y.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China(Grant No.U1964206), National Natural Science Foundation of China(Grant No. 51908018) and National Natural Science Foundation of China(Grant No. 51878020).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets generated and analysed during the current study are available from the corresponding author on reasonable request.

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

References

1. Li, X.; Pan, G.; Wu, Z.; Qi, G.; Li, S.; Zhang, D.; Zhang, W.; Wang, Z. Prediction of urban human mobility using large-scale taxi traces and its applications. *Front. Comput. Sci.* **2012**, *6*, 111–121. [\[CrossRef\]](#)
2. Deng, D.; Shahabi, C.; Demiryurek, U.; Zhu, L.; Yu, R.; Liu, Y. Latent Space Model for Road Networks to Predict Time-Varying Traffic. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 1525–1534. [\[CrossRef\]](#)
3. Tong, Y.; Chen, Y.; Zhou, Z.; Chen, L.; Wang, J.; Yang, Q.; Ye, J.; Lv, W. The simpler the better: A unified approach to predicting original taxi demands based on large-scale online platforms. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017; pp. 1653–1662.
4. Wu, F.; Wang, H.; Li, Z. Interpreting traffic dynamics using ubiquitous urban data. In Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Burlingame, CA, USA, 31 October–3 November 2016; pp. 1–4.
5. Fu, R.; Zhang, Z.; Li, L. Using LSTM and GRU neural network methods for traffic flow prediction. In Proceedings of the 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC 2016), Wuhan, China, 11–13 November 2016; pp. 324–328. [\[CrossRef\]](#)
6. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [\[CrossRef\]](#)
7. Tedjopurnomo, D.A.; Bao, Z.; Zheng, B.; Choudhury, F.; Qin, A.K. A Survey on Modern Deep Neural Network for Traffic Prediction: Trends, Methods and Challenges. *IEEE Trans. Knowl. Data Eng.* **2020**, *14*, 1. [\[CrossRef\]](#)
8. Wang, D.; Cao, W.; Li, J.; Ye, J. DeepSD: Supply-demand prediction for online car-hailing services using deep neural networks. In Proceedings of the 2017 IEEE 33rd International Conference on Data Engineering (ICDE), San Diego, CA, USA, 19–22 April 2017; pp. 243–254.
9. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [\[CrossRef\]](#)
10. Davis, N.; Raina, G.; Jagannathan, K. Grids versus graphs: Partitioning space for improved taxi demand-supply forecasts. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 6526–6535. [\[CrossRef\]](#)
11. Zhao, Z.; Chen, W.; Wu, X.; Chen, P.C.Y.; Liu, J. LSTM network: A deep learning approach for short-term traffic forecast. *IET Intell. Transp. Syst.* **2017**, *11*, 68–75. [\[CrossRef\]](#)
12. Cui, Z.; Ke, R.; Pu, Z.; Wang, Y. Deep Bidirectional and Unidirectional LSTM Recurrent Neural Network for Network-wide Traffic Speed Prediction. *arXiv* **2018**, arXiv:1801.02143.
13. Yu, R.; Li, Y.; Shahabi, C.; Demiryurek, U.; Liu, Y. Deep learning: A generic approach for extreme condition traffic forecasting. In Proceedings of the 2017 SIAM International Conference on Data Mining (SDM), Houston, TX, USA, 27–29 April 2017; pp. 777–785. [\[CrossRef\]](#)
14. Yao, H.; Wu, F.; Ke, J.; Tang, X.; Jia, Y.; Lu, S.; Gong, P.; Li, Z.; Ye, J.; Chuxing, D.; et al. Deep multi-view spatial-temporal network for taxi demand prediction. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32, pp. 2588–2595.

15. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [[CrossRef](#)]
16. Lv, Z.; Xu, J.; Zheng, K.; Yin, H.; Zhao, P.; Zhou, X. Lc-rnn: A deep learning model for traffic speed prediction. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18), Stockholm, Sweden, 13–19 July 2018; pp. 3470–3476.
17. Lai, G.; Chang, W.C.; Yang, Y.; Liu, H. Modeling long- and short-term temporal patterns with deep neural networks. In Proceedings of the the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor, MI, USA, 8–12 July 2018; pp. 95–104. [[CrossRef](#)]
18. Jogin, M.; Mohana; Madhulika, M.S.; Divya, G.D.; Meghana, R.K.; Apoorva, S. Feature Extraction using Convolution Neural Networks (CNN) and Deep Learning. In Proceedings of the 2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, India, 18–19 May 2018; pp. 2319–2323.
19. 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, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
20. Dai, B.; Wipf, D. Diagnosing and enhancing VAE models. *arXiv* **2019**, arXiv:1903.05789.
21. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
22. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 3104–3112.
23. Cai, L.; Janowicz, K.; Mai, G.; Yan, B.; Zhu, R. Traffic transformer: Capturing the continuity and periodicity of time series for traffic forecasting. *Trans. GIS* **2020**, *24*, 736–755. [[CrossRef](#)]
24. Jaderberg, M.; Simonyan, K.; Zisserman, A. Spatial transformer networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 2017–2025. [[CrossRef](#)]
25. Yu, H.; Wu, Z.; Wang, S.; Wang, Y.; Ma, X. Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks. *Sensors* **2017**, *17*, 1501. [[CrossRef](#)] [[PubMed](#)]
26. Ma, J.; Chan, J.; Ristanoski, G.; Rajasegarar, S.; Leckie, C. Bus travel time prediction with real-time traffic information. *Transp. Res. Part C Emerg. Technol.* **2019**, *105*, 536–549. [[CrossRef](#)]
27. Zhu, L.; Laptev, N. Deep and confident prediction for time series at uber. In Proceedings of the 2017 IEEE International Conference on Data Mining Workshops (ICDMW), New Orleans, LA, USA, 18–21 November 2017; pp. 103–110.
28. Lütkepohl, H. Vector autoregressive models. In *Handbook of Research Methods and Applications in Empirical Macroeconomics*; Edward Elgar Publishing: Cheltenham, UK, 2013.
29. Ma, J.; Theiler, J.; Perkins, S. Accurate on-line support vector regression. *Neural Comput.* **2003**, *15*, 2683–2703. [[CrossRef](#)] [[PubMed](#)]