



Article Convolutional Long Short-Term Memory Two-Dimensional Bidirectional Graph Convolutional Network for Taxi Demand Prediction

Yibo Cao^{1,2,*}, Lu Liu^{1,2} and Yuhan Dong^{1,*}

- ¹ Shenzhen International Graduate School, Tsinghua University, Shenzhen 518000, China
- ² Shenzhen Urban Transport Planning Center, Shenzhen 518057, China
- * Correspondence: caoyb20@mails.tsinghua.edu.cn (Y.C.); dongyuhan@sz.tsinghua.edu.cn (Y.D.)

Abstract: With the rise of the online ride-hailing market, taxi demand prediction has received more and more research interest in intelligent transportation. However, most traditional research methods mainly focused on the demand based on the original point and ignored the intention of the passenger's destination. At the same time, many forecasting methods need sufficient investigation and data processing, which undoubtedly increases the complexity and operability of forecasting problems. Therefore, we regard the current taxi demand prediction as an origin–destination problem in order to provide more accurate predictions for the taxi demand problem. By combining a spatial network based on graph convolutional network (GCN) and a temporal network of convolutional long short-term memory (Conv-LSTM), we propose a new spatial-temporal model of Conv-LSTM two-dimensional bidirectional GCN (CTBGCN) to uncover the potential correlation between origin and destination. We utilize the temporal network for effective temporal information and the spatial network of multi-layers to get the implicit origin–destination correlation. Numerical results suggest that the proposed method outperforms the state-of-the-art baseline and other traditional methods.

Keywords: taxi demand forecasting; origin-destination; bidirectional convolution

1. Introduction

As an important part of the current transportation system, taxis play a major role in the demand for urban transport capacity. Nowadays, the chief problem in the taxi market is that the distribution of taxis is unbalanced. There are lots of taxis at airports, railway stations, and other transportation hubs or commercial centers, causing congestion and vacancy. If there is no reservation, people always have to wait for a long time in marginal areas, which causes a huge waste of resources and human consumption. Therefore, measuring the taxi demand of a city properly is significant for the building of intelligent transportation systems (ITS) [1]. An effective taxi demand prediction can both better allocate urban transportation resources and reduce the situation of empty taxis, increasing the income of drivers [2].

Taxi demand prediction is a classical traffic task that attracts many researchers to the challenge. Generally, researchers predict future demand based on historical demand data and some auxiliary information such as weather and accidents [3,4]. Compared to some traditional traffic forecasting problems, taxi demand prediction is more difficult because of the irregular requirements in different zones at different time intervals. Furthermore, taxi operation does not have a clear route or station, nor an exact destination for every trip.

Because of these difficulties, most prior works focused on region-level forecasting and predicted only regional inflow and outflow. The region-level forecasting can be transformed into traffic flow predictions, such as some notable works and references therein [5–7]. Other researchers mainly focused on origin–destination (OD)-level predictions, which aim to predict the order quantity from region to region [8–10]. Taking *N* regions as an example, for each time interval, the former methods have 2N features, which indicates that each



Citation: Cao, Y.; Liu, L.; Dong, Y. Convolutional Long Short-Term Memory Two-Dimensional Bidirectional Graph Convolutional Network for Taxi Demand Prediction. *Sustainability* **2023**, *15*, 7903. https:// doi.org/10.3390/su15107903

Academic Editor: Armando Cartenì

Received: 30 January 2023 Revised: 2 March 2023 Accepted: 21 March 2023 Published: 11 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/).

2 of 14

region has an inflow and an outflow. But the latter strategies have N^2 features, which have one-to-one correspondence among the regions. Both types of methods have been used in the corresponding scenarios. Obviously, the region-level prediction pays more attention to the change in one single zone in demand. It does not consider the internal links between regions, but rather cares about the needs of the destination and ignores the impact of the destination. This indicates two implicit problems: (1) The forecast will exacerbate the taxi imbalance, causing taxis to aggregate to these areas with high demand. Simultaneously, too many vehicles can affect traffic near high-demand areas due to the interconnections of the urban transportation network, especially in traffic arteries. (2) Region-level prediction cannot capture the deep links of inflow and outflow among the regions. Since the prediction at the regional level only focuses on the total inflow and outflow of region, it does not really reveal the relationship between outflow and inflow in detail therefore the results are not reliable enough. For example, if one region has a high demand all the time, region-level prediction cannot know whether the demand is from all regions equally or one region particularly.

Corresponding to the above prediction methods, OD-level prediction makes more sense in the taxi industry. It reveals the change in passenger traffic volume between different regions, making it more reasonable to dispatch taxis throughout the whole city and balance taxi distribution. Therefore, we aim to build an OD-level method to solve the problem of taxi demand prediction [11]. In the early years, prediction methods were based on traditional machine learning schemes (see [12-14] and references therein). They mainly applied on a linear combination of historical information and superposition of random noise. Without spatial information, these results are insufficient and therefore are suitable to simple situations as necessary. At present, deep learning (DL) provides more effective ways to combine the features; most DL-based methods investigate prediction problems and extract features from the spatial and temporal dimensions. Convolutional neural networks (CNNs) [15] and graph convolutional neural networks (GCNs) [16,17] are the mainstream neural networks that capture spatial information. For the OD-level prediction, CNNs lose remote spatial information and does not deal well with the irregular spatial segmentation. However, even though GCNs consider the spatial correlations with diverse nodes implicitly, for N^2 features, a GCN either selects a higher complexity from o(N) to $o(N^2)$ in the graph size or chooses only a single node that is graphically convolved with all other nodes as one batch. Therefore, the traditional graph convolution cannot deal with the prediction problem well at the OD level. The temporal dimension usually makes use of recurrent neural network (RNN) [18], long short-term memory (LSTM) [19], gated recurrent unit (GRU) [20], and other relative networks [21–23]. Inspired by previous work, we propose a novel taxi demand prediction method of Conv-LSTM two-dimensional bidirectional GCN (CTBGCN), including two main sections about spatial and temporal contents respectively. First, considering the temporal characteristics within the spatial structure, we use Conv-LSTM [21] to fully retain the global spatial information in the time interval. Then, we choose the two-dimensional GCN (2DGCN) inspired from geometric matrix completion (GMC) [22] and multi-perspective GCN (MPGCN) [23] to extract the connections behind these origins and destinations. We utilize two adaptive graphs from their respective views to strengthen their disparate functions and alternate the graph construction in 2DGCN in particular. Extensive experiments suggest that the proposed CTBGCN outperforms the state-of-the-art method and other traditional approaches.

The contributions of this work are summarized as follows:

- We build a neural network to connect two regions to get useful information from the demand distribution of the destination.
- We design two GCNs to mine the implicit information of the origin and the destination individually, which can help balance the influence of each dimension in the results.
- Since the traditional relationship defined by static graphs may be not accurate in the real world, we use dynamic graphs to describe each region's relationship. This method not only solves the intricate graph correlations that need much previous data

to support and calculate but also reduces pre-defined difficulties if the raw data are particularly anomalous.

The rest of this paper is organized as follows. Related works are summarized in Section 2. We present the problem definition and the construction details of the proposed model, respectively, in Sections 3 and 4. The experimental settings and results are analyzed and compared in Section 5. Finally, our conclusions and future work are summarized in Section 6.

2. Related Works

2.1. Traditional Methods

In the early years, the first research into taxi demand prediction was to establish a supply and demand relationship based on road conditions and some historical data to guide taxi operation under the policy [24]. Afterward, forecasting the starting order number became mainstream. Some classical statistical methods such as historial average (HA) [12] and some traditional machine learning methods such as linear regression (LR) [13] were used in this task. Most of these methods only use a linear combination of historical data or make some regression predictions. They may include some mining of time interval information, but offer few spatial-level characteristics. At the OD level, OD problems have been investigated for about three decades mainly by three major types of approaches of generalized least squares (GLS) [25], maximum likelihood estimation [26], and bayesian methods [27]. The relationship between the historical flow and the predicted flow was constructed from the perspective of regression and probability distribution in these approaches. The optimization process was to minimize the difference between the historical and predicted flows, which has been well derived mathematically while suffer from obvious disadvantages. On the one hand, it is very difficult to deal with some spatial-temporal prediction since spatial information is completely ignored. On the other hand, the predicted results are often not accurate enough due to the simple combining of features.

2.2. Deep Learning Methods

In the field of transportation, deep learning has made remarkable achievements in traffic flow, speed, and demand problems. For taxi demand, multi-layer perception (MLP) was the first neural network used for the taxi requirements of Tokyo [28], which is an original network in today's view. With the development of neural network, most researchers utilized the convolutional construction [18] for the extraction and aggregation of spatial characteristics and the temporal neural network [19] for temporal extraction. In the regionlevel predictions, fusion convolutional LSTM network (FCL-Net) [5] fuses convolutional LSTM layers, standard LSTM layers, and convolutional layers, which is a new attempt for dealing with the embedding of heterogeneous models. Deep multi-view spatial-temporal network (DMVST-Net) [6] considers the spatio-temporal relationship of taxi demand by utilizing a serial structure of CNN and LSTM and introduces semantic similarity of different regions to measure the global association. Spatio-temporal multi-GCN (ST-MGCN) [7] chooses a multi-graph GCN with channel-wise attention in the prediction. It creatively utilizes GCN and MLP as the gate for RNN to accomplish the mixture in a spatio-temporal network instead of a serial one. Spatial-temporal graph to sequence model (STG2Seq) [29] models a multi-step city ride demand prediction and uses a multi-layer convolutional structure to capture both spatial and temporal associations. Coupled layer-wise graph convolution (CCRNN) [30] also submits a multi-layer network in which each layer has an adaptive adjacent graph to learn. It is assumed that each layer may have a different semantic similarity. Particularly with the rise of online car-hailing, the demand problem for taxis and online car-hailing needs more refined prediction results. In this field, gridembedding based multi-task learning (GEML) [4] early attempts to introduce OD-level prediction. It sets multi-tasks in OD, origin, and destination perspectives with the main task belonging to the OD problem. Contextualized spatial-temporal network (CSTN) [11] supposes a three-dimensional CNN (3DCNN) and Conv-LSTM networks, which suit the

OD features in form. It also utilizes an adaptive graph implicitly in capturing global relationship. Spatio-temporal encoder-decoder residual multi-GCN (ST-ED-RMGC) [31] designs an encoder-decoder model whose encoder section includes spatial dependence and temporal dependence. The spatial dependence is encoded by residual multi-graph convolution (RMGC), and the temporal dependence is encoded by LSTM. The two dependences concentrate together and are decoded into one vector by RMGC. All of these methods ignore destination extraction. At the OD level, they are more complicated and inefficient. Motivated by these prior works, we proposed a spatio-temporal method of CTBGCN to predict the taxi demand in OD-level to solve the difficulty of graph definition and capture the implicit information in the perspective of destination.

3. Preliminaries and Problem Statement

3.1. OD Graph

A graph structure is often denoted as G = (V, E, A), where V stands for the whole set of OD pairs in the graph instead of the node features as usual, and N = |V| represents the quantity of OD pairs at each time interval. $E \in R^N$ is the set of the graph edges. $A \in R^{N \times N}$ denotes the adjacency matrix among nodes in the graph. In particular, the adjacency matrix, which consists of distance, proximity, and other physical information in the real world, may not reflect the adjacent connection correctly. For example, airports and railway stations are generally far apart in a city, while the demand between these two regions is still high.

3.2. OD Features

We simply use X^t as the features for an integer time interval t in minutes, e.g., 1 January 2023 0:00–0:30, where X means the set of whole features. Mathematically, $X^t \in R^{N \times N \times F}$ has three dimensions, but in general, traffic prediction is $X^t \in R^{N \times F}$, where N and F are the numbers of nodes and features, respectively. Therefore, The dimensions of $N \times N$ need to be well processed in mining feature information. From the basic requirement of characteristics, $X_{i,j}^t$ is the order number from the *i*-th region to *j*-th region, which means the F dimension has only one element to simplify the combination of the dataset.

3.3. Problem Definition

We denote the inputs and outputs as *X* and *Y*, respectively. The historical information is usually taken from *n* time intervals before the predicted time interval *t* or is based on the information from the last corresponding time interval of the week or month. Furthermore, heterogeneous information such as weather is not added in the generation of the prediction model. Finally, the process is to build a function between $X \in \mathbb{R}^{n \times N \times N \times F}$ and $Y \in \mathbb{R}^{N \times N \times F}$. It can be described in detail as:

$$\{X^{t-1}, X^{t-2}, ..., X^{t-n}\} \quad \underbrace{\mathcal{F}}_{} \quad \{Y^t\} \tag{1}$$

where
$$X^{t-1} = \{X_{1,N}^{t-1}, ..., X_{i,N}^{t-1}, ..., X_{1,N}^{t-1}\}$$
 and $X_{i,N}^{t-1} = \{X_{i,1}^{t-1}, ..., X_{i,j}^{t-1}, ..., X_{i,N}^{t-1}\}$.

4. The Proposed CTBGCN Model

4.1. CTBGCN Framework

CTBGCN is a spatio-temporal network for OD prediction that contains a Conv-LSTM section to process the temporal information according to the graph structure. This satisfies the spatial structure implied in the node-to-node OD relation. Then, 2DGCN integrates spatial features from the original perspective and the destination perspective. On the one hand, 2D graph convolution is carried out from the starting point and the ending point, respectively. On the other hand, in the process of graph convolution, we interactively convolve two graphs with the starting point and ending point. The information from the different initial relationship is retained. The process of information fusion is more in line with previous logic. Three layers of TBGCN are used to increase the receptive field. In the



end, we use MLP to fuse spatial information. The overall architecture of CTBGCN and the detailed TBGCN are shown in Figures 1 and 2. The pseudocode is listed in Algorithm 1.

Figure 1. The overall framework of the CTBGCN.



Figure 2. The architecture of the TBGCN.

Algorithm 1 CTBGCN Algorithm

Input: *x*: the OD inputs, shape: (n, N, H, W); *n*: the *n* time intervals; *N*: the number of regions; *H*: the height of regions; *W*: the width of regions; $(H \times W = N)$ actually; *C*: the channel number;

Output: *y*: the OD predictions, shape: (*N*, *H*, *W*);

- 1: transpose and reshape $x : (n \times N \times H \times W) \rightarrow (N \times n \times H \times W) \rightarrow (N \times n \times C_1 \times H \times W)$, $C_1 = 1$ is the initial channel number;
- 2: $x_{embed} = Conv LSTM(x)[-1]$, shape: (N, C₂, H, W), C₂ = 64 is the channel number;
- 3: reshape and transpose x_{embed} : $(n \times C_2 \times H \times W) \rightarrow (N \times C_2 \times N) \rightarrow (N \times N \times C_2);$
- 4: $x_o = x_{embed}$; x_d : transpose x_{embed} dimension 0 and 1;
- 5: build two graphs A_o and A_d from E_o and E_d ;
- 6: $x_o = TBGCN(x_o, A_o, A_d); x_d = TBGCN(x_d, A_d, A_o);$
- 7: concatenate x_o and x_d ;
- 8: $y_{pred} = MLP(MLP([x_o, x_d]))$
- 9: return y_{pred};

4.2. Temporal Section

Every region is associated with all other regions at every time due to the randomness of taxi demand. However, the temporal neural networks, such as RNN, LSTM and etc., can only deal with the temporal evolution of a single region. Therefore, the influence of the neighboring node is lost in the time interval. Unlike these examples, in this paper, we propose to adopt Conv-LSTM to replace the parameter method in LSTM for learning the input X_t and hidden state H_t , which can introduce the form of weight sharing to retain spatial structure information for the subsequent graph convolution. The two sets of time information are memory state and hidden state, which interact in the model. The process of Conv-LSTM can be expressed as:

$$i_{t} = \sigma(W_{xi} * \mathcal{X}_{t} + W_{hi} * \mathcal{H}_{t-1} + W_{ci} \odot \mathcal{C}_{t-1} + b_{i})$$

$$f_{t} = \sigma(W_{xf} * \mathcal{X}_{t} + W_{hf} * \mathcal{H}_{t-1} + W_{cf} \odot \mathcal{C}_{t-1} + b_{f})$$

$$\mathcal{C}_{t} = f_{t} * \mathcal{C}_{t-1} + i_{t} \odot \tanh(W_{xc} * \mathcal{X}_{t} + W_{hc} * \mathcal{H}_{t-1} + b_{c})$$

$$o_{t} = \sigma(W_{xo} * \mathcal{X}_{t} + W_{ho} * \mathcal{H}_{t-1} + W_{co} \odot \mathcal{C}_{t} + b_{o})$$

$$\mathcal{H}_{t} = o_{t} \odot \tanh(\mathcal{C}_{t})$$

$$(2)$$

where i_t , f_t , and o_t are the gates of memory, forgotten, and output. All W are learnable parameters e.g., W_{xi} is the weight of X_t in the current memory gate i_t , W_{xo} is the weight of X_t in the output gate o_t . * and \odot are the operations of convolution and Hadamard product, respectively. C_t represents the memory of time intevals. \mathcal{H}_t is the final output. f_t selectively retains the memory of the previous moment, while i_t memorizes the current input and the hidden information of the previous moment. Finally, the output is obtained by combining the current memory, last output, and input. Its structure is similar to that of LSTM, but the way of learning the weights is convolution, so that the potential spatial information of the data can be preserved to a large extent. Many areas of taxi data are rectangular areas represented by multiple nodes, so Conv-LSTM can generally extract OD information on the temporal side. The temporal section acts as an encoder to increase the number of features, F', corresponding to F.

4.3. Spatial Section

4.3.1. Adjacency Matrix Definition

A graph is defined by a group of learnable parameters that aims to situate the OD data implicitly. From the description mentioned in Section 1, the relationship between origin–destination is not clear, and it is hard to measure the real functional region influence. Therefore, we take two graphs A_o and A_d from the origin and destination perspectives by initiating each graph as a learnable parameter. They are obtained by

$$A_o = \operatorname{softmax}(\operatorname{ReLU}(E_o E_o^T)) \tag{3}$$

$$A_d = \operatorname{softmax}(\operatorname{ReLU}(E_d E_d^T)).$$
(4)

where E_o and E_d are learnable parameters to describe the intensity of each node at one interval from origin perspective and destination perspective, respectively. $E_o \in \mathbb{R}^{N \times S}$ is based on the node number and the length of the sequence. Therefore, $E_o E_o^T$ can be interpreted as the similarity of each node in different intervals. It can simply be simply referred to as a kind of attention. At the same time, the result of $E_o E_o^T$ satisfies the requirement that the Laplace matrix is symmetric and semi-definite. ReLU(·) and softmax(·) are both activation functions. ReLU(·) attempts to make the attention at least zero to show a little relationship between the nodes instead of a negative result, matching the next activation function's input. Softmax(·) is used for normalizing the effect of their attention.

4.3.2. TBGCN

The standard form of graph convolution is as follows:

$$X_{i+1} = \sigma \Big(U g_{\theta}(\Lambda) U^T X_i \Big)$$
⁽⁵⁾

where *U* and Λ are the eigenvectors and eigenvalue matrices. $g_{\theta}(\Lambda)$ is the learnable convolutional kernel. The complexity of convolution depends on the eigenvalue decomposition

of the Laplace matrix and learning its convolution kernel in a GCN. Defferrard et al. [32] proposed a Chebyshev GCN to avoid eigenvalue decomposition by transferring the convolution kernel to a polynomial representation. The Chebyshev GCN is given as follows:

$$X_{i+1} = \sigma(Ug_{\theta}(\Lambda)U^T X_i) \to X_{i+1} = \sigma\left(U\sum_{k=0}^{K-1} \alpha_k \Lambda^k U^T X_i\right)$$
(6)

based on:

$$L^2 = U\Lambda U^T U\Lambda U^T = U\Lambda^2 U^T \tag{7}$$

$$\to X_{i+1} = \sigma\left(\sum_{k=0}^{K-1} \alpha_k L^k X_i\right) \tag{8}$$

Therefore, we use T_K to represent the Chebyshev polynomial. We can choose the order of neighbors by the magnitude of *K*. For instance, K = 1 means we only choose the first-order neighbor of a node to join in the convolution. Finally,

$$\rightarrow X_{i+1} = \sigma\left(\sum_{k=0}^{K-1} \beta_k T_k(L) X_i\right) \tag{9}$$

where β_k replaces the position of α_k as the learnable parameter. It is the most common modality of the Chebyshev GCN in many traffic tasks, such as ST-MGCN [7]. We introduce the Chebyshev GCN to combine the origin and destination convolution. This increases the kernel of convolution, and its initial definition is:

$$X_{i+1} = \sigma \left(\sum_{m=0}^{M-1} \sum_{k=0}^{K-1} \beta_{k,m} T_k(L) X_i T_m(L^T) \right)$$
(10)

where *M* is the magnitude of L^T . Considering the different modes of origin and destination, we creatively change the *L* and L^T to L_o and L_d . L_o and L_d are calculated from the adjacency matrix of Section 4.3.1. The Chebyshev GCN transforms as:

$$X_{i+1} = \sigma \left(\sum_{m=0}^{M-1} \sum_{k=0}^{K-1} \beta_{k,m} T_k(L_o) X_i T_m(L_d) \right)$$
(11)

The above method more fully considers the change in relative attention of nodes in OD. In addition, inspired by CSTN, we transpose the node information learned from the temporal network on the OD dimension and then carry out data mining from the perspective of the destination. Bidirectional data mining can improve the results of the ablation experiment.

4.4. Fusion Section

The fusion section is composed of two layers of MLP to fuse the two groups of GCN results and achieve non-linearity. This section first concatenates the results and then puts them into the decoder MLP to decrease the number of features to 1 as the output.

5. Experiments and Results

5.1. Dataset

The dataset was collected from the taxi OD statistics of the New York City Taxi Commission. This includes orders for New York City yellow and green cabs from 2009 to 2022 dataset (accessed on 29 January 2023). It contains the starting and ending position information of the taxi and start-stop time. Figure 3 shows the order heat map on one random day between 0:00 and 0:30 in 2014. In detail, we use the dataset [11], which includes the OD data for 2014. The whole dataset uses half an hour for each time interval,

mainly focusing on the study of taxi OD on Manhattan Island. In the spatial dimension, the geographical space of the island is divided into 15×5 areas, each of which is square. Figure 4 presents the segmentation method in a real map and the raw data in the central region. Thus, there are 17,520 ($365 \times 2 \times 24$) pieces of data in total for the whole year. Each piece of data contains OD information for about 75 areas.



Figure 3. The OD demand between 0:00–0:30 within one day.



Figure 4. The (**left**) part is our split method. The (**right**) part shows the data in the central region at 12:00 on 1 January 2014. The shade of color represents different order quantity.

5.2. Evaluation Metrics and Experiment Settings

• Evaluation metrics: To evaluate the performance of the proposed model, we adopt performance metrics of mean square errors (MSE), root mean square errors (RMSE), and mean absolute percentage errors (MAPE). *OD_RMSE* and *OD_MAPE* are used for test evaluation at the OD-level. *O_RMSE* and *O_MAPE* refer to the comparisons at the region level in RMSE and MAPE. *OD_MAE* is regarded as the loss function to prompt the network convergence. Their mathematical formulas are listed here:

$$OD_MSE = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} (Y_{ij}^P - Y_{ij}^T)^2$$

$$OD_RMSE = \sqrt{\frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} (Y_{ij}^P - Y_{ij}^T)^2}$$

$$OD_MAPE = \frac{100\%}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} |\frac{Y_{ij}^P - Y_{ij}^T}{Y_{ij}^T}|$$

$$O_RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (Y_i^P - Y_i^T)^2}$$

$$O_MAPE = \frac{100\%}{N} \sum_{i=1}^{N} |\frac{Y_i^P - Y_i^T}{Y_i^T}|$$
(12)

where Y_{ij}^{p} represents the prediction of order number from the *i*-th region to the *j*-th region, Y_{ij}^{T} represents the ground truth, and Y_{i}^{p} and Y_{i}^{T} are for the region level.

 Data preparation: We make use of all of the data in the prediction and set the ratio of the training, verification, and test to 4:1:1 in chronological order. According to the Min-Max method, the data transformed in the range of [-1,1], which can be expressed as:

$$X' = 2X / X_{max} - 1. (13)$$

After the transformation, we utilize the demands of five continuous intervals to predict the demand of the next interval. For instance, the data during 4:00 to 6:30 are used to forecast the demand during 6:30 to 7:00 within one day.

• Experiment settings: All experiments were run on one GeForce RTX 2080 Ti GPU and one Intel(R) Xeon(R) CPU E5-2678 v3 @ 2.50 GHz. The *k*-th order of the Chebyshev net is set as 3. We set 500 epochs to train the proposed model and set the early stopper at 50 to ensure the best results. The optimized method is the adaptive moment estimation (Adam). The initial learning rate is 0.001. After 100 epochs, we reduce the learning rate by half if the loss does not decrease within 5 epochs. Finally, we set the batch size at 16 because of the limit of video memory.

5.3. Methods for Comparison

- HA [12]: Predict the future needs by averaging the historical demands. We utilize the same data from *S* time intervals to average the taxi demands of the previous *n* time intervals.
- Lasso Regression [13]: Lasso Regression is a linear regression method with ℓ1-norm regularization. The data used are the same as in the HA method above.
- LSTM [19]: LSTM is a temporal neural network for many time-interval problems. Three layers are added, and the hidden dimensions are 16, 32, and 1.
- MLP: MLP is a classical deep learning method. We set the hidden neurons at 64, 32, 16, and 1.
- ST-ResNet [33]: ST-ResNet is a residual neural network framework that models the time distance, cycle, and trend attributes of crowd flow.
- CSTN [11]: CSTN is an OD predicting method. It makes use of the 3DCNN, Conv-LSTM, and a global correlation context section from global and local views.

5.4. Comparisons

Table 1 shows RMSE and MAPE performance for various methods. The proposed method achieves the best results in all adopted metrics. Three main conclusions are summarized by analyzing the experiments and results:

- CTBGCN achieves the best performance in metrics. Compared with CSTN, there is an obvious improvement in O_MAPE at the region level, with a relative performance improvement of 4.5%. Other metrics improve slightly, with a relative improvement of just 0.2% in OD_RMSE and 0.14% in OD_MAPE. However, other baselines have gaps in terms of both RMSE and MAPE since they all ignore spatio-temporal correlations.
- The extensibility of the proposed model is wonderful. The model do not need any predefined additional information in the prediction process in addition to the necessary OD information.
- The proposed model has a excellent generalization in long-term predictions. Many
 other methods do not make sense for long time intervals. There is no validation set or
 the validation set time is very short in methods such as CSTN and ST-ED-RMGC. We
 allocate two months to the verification set in our experiment, which means our model
 needs to fit the data after two months correctly. It proves that the proposed model has
 great robustness even in the face of some weather and seasonal changes.

Table 1. RMSE and MAPE performance for various OD-level and region-level taxi demand prediction methods.

Methods	OD_RMSE	OD_MAPE	O_RMSE	O_MAPE
HA	1.893	35.46%	54.33	47.59%
Lasso	1.652	33.85%	33.00	34.89%
MLP	1.665	34.35%	34.79	28.84%
LSTM	1.618	33.55%	33.86	40.18%
ST-ResNet	1.380	28.53%	22.43	24.16%
CSTN	1.322	27.39%	19.85	18.48%
CTBGCN (ours)	1.318	27.25%	18.95	18.16%

5.5. Forecating Shows

We randomly select the prediction results of one piece of data in the test set in the daytime. Figure 5 is the comparison of these data (data size: 75×75). As can be seen, the areas of high demand are mainly in the central location. The proposed model achieves a excellent performance of predicting overall demand. The MAE of more than 2 is forecast at only 9.2% at the OD level, which means that the error of most predictions is limited. Specifically, we choose the highest demand area of the year to demonstrate the superiority of the model. Figure 6 shows the OD demand from the highest demand area to other regions at that interval. At the highest demand of 44, the proposed method has a very close prediction of 44.35 as shown in this figure, which shows that CTBGCN works at predicting high demand. We measure the top 20 high-demand areas within a year and show the metrics for these areas in Table 2. All regions utilize the data whose OD is from these regions to all regions (75 regions) and twenty regions use the data within themselves. We can see that the model keeps up a stable performance in high-demand regions compared with the whole dataset.



Figure 5. The prediction, actual demand, and the comparison in the middle of one day in the test dataset.



Figure 6. The prediction, actual demand, and the comparison of the highest demand area of the year in Figure 5.

Table 2. RMSE and MAPE performance for various OD-level and region-level of all regions and20 regions.

Conditions	OD_RMSE	OD_MAPE	O_RMSE	O_MAPE
all regions	1.326	27.44%	19.07	21.82%
20 regions	2.161	26.33%	31.74	19.56%

5.6. Time Consumption

Table 3 lists the average consumption time of CSTN and CTBGCN in one epoch at the same device. To be fair, the total time for each epoch to train, back-propagate, and save the model is counted. CTBGCN consumes a relatively short time and speculate that CSTN may consume too much time in a 3D convolution operation. Obviously, the proposed approach has a significant advantage in terms of time consumption.

Table 3. Time consumption for CSTN and CTBGCN.

Methods	Time Consumption/Epoch		
CSTN	1013 s		
CTBGCN (ours)	420 s		

5.7. Ablation Experiments

Ablation experiments are commonly employed to evaluate the effectiveness of proposed models. In this study, six distinct model variations are designed and tested to validate the proposed approach. Specifically, LSTM and Conv-LSTM are compared in terms of their ability to extract temporal information. Furthermore, alterations are made to the spatial section of the model, including changes to the graph composition, graph correlation method, and bidirectional construction, to assess the contributions of these components. Additionally, a CNN implemented prior to the temporal network to investigate the influence of physical neighbor nodes on the OD demand. The experimental results are presented in Table 4.

Methods	OD_RMSE	OD_MAPE	O_RMSE	O_MAPE
LSTM+ 1 layer TGCN	1.434	29.57%	22.91	21.46%
LSTM+ 3 layers TGCN	1.391	28.86%	21.30	19.61%
LSTM+ 3 layers TBGCN	1.362	28.02%	21.21	23.71%
LSTM+ 3 layers TBGCN (static graph)	1.413	29.65%	23.28	23.78%
CL+ 3 layers TBGCN	1.320	27.34%	19.06	19.04%
CNN+ CL+ 3 layers TBGCN (cor)	1.322	27.31%	19.23	18.31%
CL+ 3 layers TBGCN (cor) (ours)	1.318	27.25%	18.95	18.16%

Table 4. Ablation experiments for various configurations.

The changes of the ablation experiments are listed here. First, one layer spatial section and three layers are set to compare the function of spatial stack. More spatial layers can retain more spatial information. And then, we compared the different structures of spatial section. TBGCN is the proposed spatial section which has a better performance compared with the spatial network TGCN, which captures the information from origin perspective alone. Furthermore, the graph structures of static and dynamic are employed to show the effect of the interactive graph. The static graph indicates a physical adjacent graph where $\alpha_{ij} = 1$ whether the areas are neighborhoods or 0. The correlative graph usage can improve the final results at all metrics.

6. Conclusions and Future Direction

We proposed a novel CTBGCN model for taxi demand prediction. Extensive experiments were conducted on the real demand data of New York City. CTBGCN considers the influence of destination on OD problem. The GCN operation and graph structure from destination perspective are referred to better capture the correlation between destination and demand. And CTBGCN do not require the pre-definition of the physical graph information, such as adjacent matrix. This would be more flexible for prediction tasks with complex graph structure. Therefore, CTBGCN can effectively extract the implicit demand correlation between OD level nodes and achieve accurate prediction. Specially, the RMSE and MAPE of CTBGCN were 18.95 and 18.16 % respectively at region-level, outperforming the best baseline with a 4.5% improvement in regional RMSE.

There are some potential research directions in the future. First, the network cannot determine some anomalies in one time sequence because of the lack of information. The addition of multimodal information can improve the prediction accuracy. And then, the graph form in spatial section is global, and it may reckon without local characteristics. Furthermore, the temporal section of CTBGCN can only deal with the regular spatial division, which requires a more effective and general temporal feature extraction module for irregular regions.

Author Contributions: Methodology, Y.C.; data curation, L.L.; formal analysis, Y.D.; writing—original draft preparation, Y.C. and L.L.; writing—review and editing, Y.D. and L.L.; supervision, Y.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

- 1. Zhao, K.; Khryashchev, D.; Freire, J.; Silva, C.; Vo, H. Predicting taxi demand at high spatial resolution: Approaching the limit of predictability. In Proceedings of the 2016 IEEE International Conference on Big Data (Big Data), Washington, DC, USA, 5–8 December 2016; pp. 833–842. [CrossRef]
- Yang, H.; Wong, S.; Wong, K. Demand–supply equilibrium of taxi services in a network under competition and regulation. *Transp. Res. Part B Methodol.* 2002, 36, 799–819. [CrossRef]
- 3. Ma, W.; Qian, Z.S. Estimating multi-year 24/7 origin-destination demand using high-granular multi-source traffic data. *Transp. Res. Part C Emerg. Technol.* **2018**, *96*, 96–121. [CrossRef]
- Wang, Y.; Yin, H.; Chen, H.; Wo, T.; Xu, J.; Zheng, K. Origin-Destination Matrix Prediction via Graph Convolution: A New Perspective of Passenger Demand Modeling. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; Association for Computing Machinery: New York, NY, USA, 2019; KDD '19, pp. 1227–1235. [CrossRef]
- 5. Ke, J.; Zheng, H.; Yang, H.; Chen, X.M. Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach. *Transp. Res. Part C Emerg. Technol.* **2017**, *85*, 591–608. [CrossRef]
- Yao, H.; Wu, F.; Ke, J.; Tang, X.; Jia, Y.; Lu, S.; Gong, P.; Ye, J.; Zhenhui, L. Deep Multi-View Spatial-Temporal Network for Taxi Demand Prediction. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
- Geng, X.; Li, Y.; Wang, L.; Zhang, L.; Yang, Q.; Ye, J.; Liu, Y. Spatiotemporal Multi-Graph Convolution Network for Ride-Hailing Demand Forecasting. In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; AAAI Press: Palo Alto, CA, USA, 2019; AAAI'19/IAAI'19/EAAI'19. [CrossRef]
- 8. Blume, S.O.; Corman, F.; Sansavini, G. Bayesian origin-destination estimation in networked transit systems using nodal in- and outflow counts. *Transp. Res. Part B Methodol.* **2022**, *161*, 60–94. [CrossRef]
- 9. Ding, F.; Zhu, Y.; Yin, Q.; Cai, Y.; Zhang, D. MS-ResCnet: A combined spatiotemporal modeling and multi-scale fusion network for taxi demand prediction. *Comput. Electr. Eng.* **2023**, *105*, 108558. [CrossRef]
- 10. Xu, J.; Rahmatizadeh, R.; Bölöni, L.; Turgut, D. A taxi dispatch system based on prediction of demand and destination. *J. Parallel Distrib. Comput.* **2021**, *157*, 269–279. [CrossRef]
- 11. Liu, L.; Qiu, Z.; Li, G.; Wang, Q.; Ouyang, W.; Lin, L. Contextualized Spatial–Temporal Network for Taxi Origin-Destination Demand Prediction. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 3875–3887. [CrossRef]
- 12. Campbell, J.Y.; Thompson, S.B. Predicting excess stock returns out of sample: Can anything beat the historical average? *Rev. Financ. Stud.* **2008**, *21*, 1509–1531. [CrossRef]
- 13. Tibshirani, R. Regression Shrinkage and Selection via the Lasso. J. R. Stat. Soc. Ser. B-Methodol. 1996, 58, 267–288. [CrossRef]
- Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; Association for Computing Machinery: New York, NY, USA, 2016; KDD '16, pp. 785–794. [CrossRef]
- 15. Liang, X.; Xu, C.; Shen, X.; Yang, J.; Tang, J.; Lin, L.; Yan, S. Human Parsing with Contextualized Convolutional Neural Network. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 115–127. [CrossRef] [PubMed]
- Li, Y.; Yu, R.; Shahabi, C.; Liu, Y. Graph Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. *CoRR* 2017, abs/1707.01926. Available online: http://xxx.lanl.gov/abs/1707.01926 (accessed on 29 January 2023).
- Yu, B.; Yin, H.; Zhu, Z. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, International Joint Conferences on Artificial Intelligence Organization, Stockholm, Sweden, 13–19 July 2018; pp. 3634–3640. [CrossRef]
- 18. Zaremba, W.; Sutskever, I.; Vinyals, O. Recurrent Neural Network Regularization. *CoRR* **2014**, abs/1409.2329. Available online: http://xxx.lanl.gov/abs/1409.2329 (accessed on 29 January 2023).
- 19. Yu, Y.; Si, X.; Hu, C.; Zhang, J. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural Comput.* **2019**, *31*, 1235–1270. [CrossRef] [PubMed]
- Chung, J.; Gülçehre, Ç.; Cho, K.; Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *CoRR* 2014, abs/1412.3555. Available online: http://xxx.lanl.gov/abs/1412.3555 (accessed on 29 January 2023).
- 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. In Proceedings of the 28th International Conference on Neural Information Processing Systems— Volume 1, Sanur, Bali, Indonesia, 8–12 December 2021; MIT Press: Cambridge, MA, USA, 2015; NIPS'15, pp. 802–810.
- Monti, F.; Bronstein, M.M.; Bresson, X. Geometric Matrix Completion with Recurrent Multi-Graph Neural Networks. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; NIPS'17; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 3700–3710.
- Shi, H.; Yao, Q.; Guo, Q.; Li, Y.; Zhang, L.; Ye, J.; Li, Y.; Liu, Y. Predicting Origin-Destination Flow via Multi-Perspective Graph Convolutional Network. In Proceedings of the 2020 IEEE 36th International Conference on Data Engineering (ICDE), Dallas, TX, USA, 20–24 April 2020; pp. 1818–1821. [CrossRef]
- Nguyen, K.P. Demand, supply, and pricing in urban road transport: The case of Ho Chi Minh City, Vietnam. *Res. Transp. Econ.* 1999, 5, 107–154. [CrossRef]

- Cascetta, E. Estimation of trip matrices from traffic counts and survey data: A generalized least squares estimator. *Transp. Res.* Part B Methodol. 1984, 18, 289–299. [CrossRef]
- Spiess, H. A maximum likelihood model for estimating origin-destination matrices. *Transp. Res. Part B Methodol.* 1987, 21, 395–412. [CrossRef]
- Maher, M. Inferences on trip matrices from observations on link volumes: A Bayesian statistical approach. *Transp. Res. Part B Methodol.* 1983, 17, 435–447. [CrossRef]
- Mukai, N.; Yoden, N. Taxi Demand Forecasting Based on Taxi Probe Data by Neural Network. In *Intelligent Interactive Multimedia: Systems and Services*; Watanabe, T., Watada, J., Takahashi, N., Howlett, R.J., Jain, L.C., Eds.; Springer Berlin Heidelberg: Berlin/Heidelberg, Germany, 2012; pp. 589–597.
- Bai, L.; Yao, L.; Kanhere, S.S.; Wang, X.; Sheng, Q.Z. STG2Seq: Spatial-temporal Graph to Sequence Model for Multi-step Passenger Demand Forecasting. *CoRR* 2019, abs/1905.10069. Available online: http://xxx.lanl.gov/abs/1905.10069 (accessed on 29 January 2023).
- Ye, J.; Sun, L.; Du, B.; Fu, Y.; Xiong, H. Coupled Layer-wise Graph Convolution for Transportation Demand Prediction. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 2–9 February 2021; Volume 35, pp. 4617–4625.
- Ke, J.; Qin, X.; Yang, H.; Zheng, Z.; Zhu, Z.; Ye, J. Predicting origin-destination ride-sourcing demand with a spatio-temporal encoder-decoder residual multi-graph convolutional network. *Transp. Res. Part C Emerg. Technol.* 2021, 122, 102858. [CrossRef]
- Defferrard, M.; Bresson, X.; Vandergheynst, P. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. *CoRR* 2016, abs/1606.09375. Available online: http://xxx.lanl.gov/abs/1606.09375 (accessed on 29 January 2023).
- Zhang, J.; Zheng, Y.; Qi, D. Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction. CoRR 2016, abs/1610.00081. Available online: http://xxx.lanl.gov/abs/1610.00081 (accessed on 29 January 2023).

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