CACRNN: A Context-Aware Attention-Based Convolutional Recurrent Neural Network for Fine-Grained Taxi Demand Prediction

Wenbin Wu¹, Tong Liu^{1,2,3(✉)}, and Jiahao Yang¹

¹ School of Computer Engineering and Science, Shanghai University, Shanghai, China
{wenbinw,tong_liu,Jiahao_Yang}@shu.edu.cn

² Shanghai Engineering Research Center of Intelligent Computing System,
Shanghai University, Shanghai, China

³ Shanghai Institute for Advanced Communication and Data Science,
Shanghai University, Shanghai, China

Abstract. As taxis are primary public transport in metropolises, accurately predicting fine-grained taxi demands of passengers in real time is important for guiding drivers to plan their routes and reducing the waiting time of passengers. Many efforts have been paid to provide accurate taxi demand prediction, and deep neural networks are leveraged recently. However, existing works are limited in properly incorporating multi-view taxi demand predictions together, by simply assigning fixed weights learned by training to the predictions of each region. To solve this problem, we apply the attention mechanism for leveraging contextual information to assist prediction, and a context-aware attention-based convolutional recurrent neural network (CACRNN) is proposed. Specially, we forecast fine-grained taxi demands with considering multi-view features, including spatial correlations among adjacent regions, short-term periodicity, long-term periodicity, and impacts of external factors. Local convolutional (LC) layers and gated recurrent units (GRUs) are utilized to extract the features from historical records. Moreover, a context-aware attention module is employed to incorporate the predictions of each region with considering different features, which is our novel attempt. This module assigns different weights to the predictions of a region according to its contextual information such as weather, index of time slots, and region function. We conduct comprehensive experiments based on a large-scale real-world dataset from New York City, and the results show that our method outperforms state-of-the-art baselines.

Keywords: Taxi demand prediction · Convolutional recurrent neural networks · Attention mechanism · Multi-view spatial-temporal feature extraction

1 Introduction

Taxis play an important role in public transportation systems, providing comfortable and convenient services to a larger amount of passengers every day, especially in metropolises like New York City. According to a survey conducted in 2016, the number of taxis is over 13,000 in New York City, and about 420,000 orders on average are completed per day. However, a major problem exists in taxi service is that the spatial-temporal imbalance between supply of drivers and demand of passengers. For example, some drivers steer empty taxis on some streets, while some passengers cannot take taxis even after a long wait on other streets. This problem leads to the increase of waiting time of passengers and the decrease of incomes of drivers.

Predicting fine-grained taxi demands in future is of great significance to solve the problem. Extracting spatial-temporal patterns from historical taxi trip records can help prediction. However, there exist several challenges to make accurate prediction. *Firstly*, taxi demands are highly dynamic, i.e., varying rapidly and randomly over time, which are determined by passengers. On the other hand, certain periodic patterns exist objectively, like high taxi demands in rush hours on weekdays. *Secondly*, the variations of taxi demands in different functional regions of a city are unlike, e.g., central business districts and residential areas. In addition, the taxi demand of a region has high correlations with other regions, especially its adjacent regions, due to the flows of passengers. *Thirdly*, taxi demands are greatly influenced by some external factors, such as weather condition, holidays and weekends. For example, many people take taxis in the early morning on the New Year's Day because of celebratory activity, which does not occur in ordinary days.

There has been a long line of studies in taxi demand prediction. Model-based methods are widely developed in the earlier works. For instance, autoregressive integrated moving average (ARIMA) and its improvements are used [6,8], via modeling taxi demand prediction problem as a time series prediction problem. Recently, deep neural networks (DNN) are introduced to predict taxi demands, in which complicated spatial-temporal correlations are extracted and external factors are used to assist prediction. For example, Xu et al. [11] propose a sequential learning framework based on long short-term memory (LSTM) network, in which instant temporal dependencies are leveraged. Convolution operation is integrated with LSTM by Yao et al. [12], and spatial correlations and temporal dependencies are both extracted. External factors are further leveraged in [2] to improve the prediction accuracy. However, these works are limited in properly incorporating multi-view features of taxi demands and external factors together, by simply assigning them fixed weights learned by training.

In this work, we propose a convolutional recurrent network model for taxi demand prediction. We first divide time into time slots and partition an urban area into regions, based on which fine-grained taxi demands are defined. Then, multi-view spatial-temporal features of taxi demands are used to perform prediction. Specially, for each region, three predictions are obtained, considering the spatial correlations and temporal dependencies among adjacent regions in

successive time slots, and the short-term and long-term periodicity with the impacts of external factors respectively. Local convolutional layers and gated recurrent units are employed in our network model. Finally, we develop a novel context-aware attention mechanism to incorporate the predictions of each region. Contextual factors are input into fully-connected layers to learn the weight assigned to each prediction, and the final prediction is calculated as the weighted sum.

The main contributions of this paper can be summarized as follows.

- We propose a convolutional recurrent network model for fine-grained taxi demand prediction. Multi-view features of taxi demands, including the spatial correlations among adjacent regions, short-term and long-term periodicity, and the impacts of external factors, are considered to perform prediction.
- We also develop a context-aware attention mechanism to incorporate the predictions of each region, by assigning them different notice. Contextual information, such as weather condition, index of time slots, and region function, are taken into account in our attention network.
- We conduct comprehensive experiments based on real-world datasets from New York City. The results show that our proposed network outperforms state-of-the-art methods.

The rest of this paper is organized as follows. Section 2 describes some key definitions and our problem formulation. The details of our designed neural network is illustrated in Sect. 3. Section 4 presents the experimental results of our model and several baselines. We review related work and conclude our paper in Sect. 5 and Sect. 6, respectively.

2 Problem Formulation

In this section, we first present some key definitions, and then formally formulate the taxi demand prediction problem.

Definition 1 (Road Network). *A road network of an urban area is composed of a set of road segments. Each road segment is associated with two terminal points (i.e., intersections of crossroads), and connects with other road segments by sharing the same terminals. All road segments compose the road network in the format of a graph.*

To formally describe fine-grained taxi demands in spatial and temporal dimensions, we discretize time into a set of equal-interval time slots, denoted by $\mathcal{T} = \{t_1, t_2, \dots, t_\tau, \dots\}$, where t_τ represents the current time slot. We also divide the whole urban area into disjoint regions based on its road network, by leveraging the map segmentation method in [14]. Each region is an irregular polygon, encompassed by several road segments. The set of regions is represented by $\mathcal{R} = \{r_1, r_2, \dots, r_N\}$, where N represents the number of regions. Based on the definitions of time slots and regions, we further present the formal definition of fine-grained taxi demands as follows.

Definition 2 (Taxi Demands). We use $X_{n,\tau}$ to represent the number of passengers with the demand of taking a taxi in region $r_n \in \mathcal{R}$ at time slot $t_\tau \in \mathcal{T}$. Then, the taxi demands at time slot t_τ are defined as $\mathbf{X}_\tau = [X_{1,\tau}, X_{2,\tau}, \dots, X_{N,\tau}]$.

Definition 3 (Taxi Tripping Records). We denote $\{tr\}$ is a set of historical taxi tripping records. Each record tr contains locations and timestamps of picking up and dropping off a passenger, which can be denoted by a tuple $tr = (tr.pl, tr.pt, tr.dl, tr.dt)$. Here, pick-up location $tr.pl$ and drop-off location $tr.dl$ are given by their latitudes and longitudes, while pick-up timestamp $tr.pt$ and drop-off timestamp $tr.dt$ are given by their dates, hours and minutes.

To predict taxi demands in future, we first dig fine-grained taxi demands in past time slots from historical taxi tripping records as defined in Definition 3. Given a dataset of taxi tripping records in an urban area, historical taxi demands in region r_n at time slot t_τ can be approximated by the number of passengers have taken a taxi, which is derived as

$$X_{n,\tau} = |\{tr | tr.pl \in r_n \wedge tr.pt \in t_\tau\}|, \quad (1)$$

where $tr.pl \in r_n$ and $tr.pt \in t_\tau$ mean that the pick-up location of record tr is in region r_n , and the pick-up timestamp is within time slot t_τ , respectively. Function $|\cdot|$ denotes the cardinality of a set.

Definition 4 (POI). A point of interest (POI) is a venue in an urban area like a shopping mall. Each POI is associated with a location and a category.

Information contained in POIs indicates the function of regions (e.g., central business districts) as well as the flows of passengers. We define a *function vector* for each region, denoted by $r_n.func$, in which each element is the number of POIs of a specific category. In addition, we also define a set of neighbours for each region composed of its adjacent regions, denoted by $r_n.neig$.

We now formulate the problem of predicting fine-grained taxi demands in the next time slot as follows.

Definition 5 (Taxi Demand Prediction Problem). Consider an urban area is divided into disjoint regions $\mathcal{R}$ by the road network. Given fine-grained taxi demands in the past time slots $\{\mathbf{X}_t | t = 1, 2, \dots, \tau\}$ extracted from historical taxi tripping records, we try to predict fine-grained taxi demands at the next time slot. The prediction is denoted as $\hat{\mathbf{X}}_{\tau+1} = [\hat{X}_{1,\tau+1}, \hat{X}_{2,\tau+1}, \dots, \hat{X}_{N,\tau+1}]$.

3 Methodology

3.1 Overview of CACRNN Model

Figure 1 provides an overview of our proposed deep neural network model, which comprises of four modules.

Instant Spatial-Temporal Module. This module is composed of a series of local convolutional (LC) layers and a gated recurrent unit (GRU), which are employed to extract spatial and temporal dependencies of taxi demands in a close period. Specially, it takes the taxi demands in o successive time slots as its input, denoted by $\mathbf{Y}^i = [\mathbf{X}_{\tau+1-o}, \mathbf{X}_{\tau+2-o}, \dots, \mathbf{X}_{\tau}] \in \mathbb{R}^{o \times N}$, and outputs a prediction of taxi demands in the next time slot $\mathbf{f}^i \in \mathbb{R}^{1 \times N}$.

Short-Term Periodic Module. This module considers the existence of short-term (e.g., a few days) periodicity in taxi demands to perform the prediction. We employ a GRU to learn the short-term periodicity, which takes a sequence of taxi demands in p periodic time slots with interval Δ_s as the input, i.e., $\mathbf{Y}^s = [\mathbf{X}_{\tau+1-p\Delta_s}, \mathbf{X}_{\tau+1-(p-1)\Delta_s}, \dots, \mathbf{X}_{\tau+1-\Delta_s}] \in \mathbb{R}^{p \times N}$. Besides, the influence of some external factors (like weather and holidays) to the periodicity is also considered in this module. We represent the features of external factors in t_{τ} as a vector $\mathbf{u}_{\tau} \in \mathbb{R}^{1 \times \omega}$, and concatenate it with taxi demands as the input of the GRU. Then, a prediction of taxi demands $\mathbf{f}^s \in \mathbb{R}^{1 \times N}$ is obtained.

Long-Term Periodic Module. This module draws the long-term (e.g., a few weeks) periodic pattern of taxi demands. Similar with the last module, a sequence of taxi demands $\mathbf{Y}^l = [\mathbf{X}_{\tau+1-q\Delta_l}, \mathbf{X}_{\tau+1-(q-1)\Delta_l}, \dots, \mathbf{X}_{\tau+1-\Delta_l}] \in \mathbb{R}^{q \times N}$ combined with features of external factors is fed to a GRU, and a prediction of taxi demands at time slot $t_{\tau+1}$ is output, denoted by $\mathbf{f}^l \in \mathbb{R}^{1 \times N}$.

Context-Aware Attention Module. We leverage an attention module to incorporate the outputs of the above modules into the final taxi demand prediction $\hat{\mathbf{X}}_{\tau+1}$, which is a novel attempt. Especially, our attention model can be interpreted as assigning different weights to the predictions of each region, according to contextual information like weather condition, index of time slots, and region function.

In the following, we provide details of each module respectively. Main notations and descriptions are summarized in Table 1, where ‘#’ represents ‘number’.

3.2 Instant Spatial-Temporal Module

The structure of this module is built based on local convolutional layers and a GRU, to extract the latent spatial correlations in adjacent regions and the temporal dependencies in a close period.

A sequence of taxi demands $\mathbf{Y}^i$ is first fed to the LC layers. In each layer, local convolutional operation is conducted, and k convolution kernels are used to extract high-dimensional spatial features. Specially, we take the l -th ($2 \leq l \leq L$) LC layer as an example to illustrate the details. We denote the input of this

Table 1. Main notations

o	# of time slots in instant spatial-temporal module
k	# of convolution kernels
L	# of LC layers
p	# of time slots in short-term periodic module
Δ_s	Interval in short-term periodic module
q	# of time slots in long-term periodic module
Δ_l	Interval in long-term periodic module

Fig. 1. Framework of our CACRNN model

layer as $\mathbf{Y}_l^i \in \mathbb{R}^{k \times o \times N}$, which also is the output of the $(l-1)$ -th LC layer. *Firstly*, for each region r_n , we construct a *sub-matrix* $\mathbf{Y}_{l,n}^i$ by rearranging some columns of $\mathbf{Y}_l^i$. Specially, we define $M = \max_{\forall n} \{|r_n.neig|\}$ and thus define $\mathbf{Y}_{l,n}^i \in \mathbb{R}^{k \times o \times (M+1)}$. For region r_n , the columns in $\mathbf{Y}_l^i$ corresponding to its neighbouring regions are chosen to be a part of $\mathbf{Y}_{l,n}^i$, as shown in Fig. 2. Besides, we pad the left vacant columns by duplicating the column in $\mathbf{Y}_l^i$ corresponding to r_n ($M+1 - |r_n.neig|$) times. *Secondly*, we conduct a convolutional operation on each $\mathbf{Y}_{l,n}^i$, respectively. Convolution kernels with size equal to $1 \times (M+1)$ are used to scan each row of $\mathbf{Y}_{l,n}^i$, and a $o \times 1$ vector is output by each kernel for r_n . We also add batch normalization (BN) after each LC layer to accelerate the training speed. By concatenating the outputs of k kernels for all regions, we get the output of the l -th LC layer, $\mathbf{Y}_{l+1}^i \in \mathbb{R}^{k \times o \times N}$, which is also the input of the $(l+1)$ -th LC layer. After L LC layers, a 1×1 convolutional operation is applied to compress high-dimensional spatial features, and a high-level representation is obtained, denoted by $\mathbf{S}^i \in \mathbb{R}^{o \times N}$.

Next, the high-level representation is fed to a GRU proposed by [3]. Specially, each row of $\mathbf{S}^i$ (denoted by $\mathbf{S}_t^i$), containing high-level spatial features at time slot $t \in [t_{\tau+1-o}, t_\tau]$, is fed to the GRU in order. The computations of this component can be represented as

$$\mathbf{h}_\tau^i = GRU(\mathbf{S}_{\tau+1-o}^i, \mathbf{S}_{\tau+2-o}^i, \dots, \mathbf{S}_\tau^i), \quad (2)$$

where $\mathbf{h}_\tau^i \in \mathbb{R}^{1 \times \kappa}$ is a high-level representation containing temporal dependencies of taxi demands among o successive time slots. Here, κ is a tunable parameter in the GRU, representing how many hidden nodes are used. Finally, a prediction of taxi demands at next time slot $t_{\tau+1}$, denoted by $\mathbf{f}^i = [f_1^i, f_2^i, \dots, f_N^i]$, is output by a fully-connected (FC) layer with input $\mathbf{h}_\tau^i$. Overall, the prediction is obtained based on recent taxi demands, considering spatial correlations among adjacent regions and temporal dependencies among successive time slots.

3.3 Short/Long-Term Periodic Module

Short-term and long-term periodicity of taxi demands are considered to perform prediction in these two modules, respectively. As shown in Fig. 1, they share the same GRU-based structure, which takes a combination of taxi demands and external factor features as its input.

Fig. 2. Structure of a LC layer

Fig. 3. Structure of attention network

External factors in temporal dimension, like weather condition and holidays/weekends, have great impact on the periodicity of taxi demands. For example, we find that the hourly variation of taxi demands in weekends or holidays is significantly different from weekdays. Besides, the peak durations of taxi demands in a rainy day and a sunny day are also different. To capture the features of external factors, we employ an embedding method [5] to transform the values of these factors at each time slot to an *external feature vector*, denoted by $\mathbf{u}_t$. This embedding method is widely used to map categorical values into a low-dimensional vector.

Then, we concatenate the sequences of taxi demands with the external feature vectors in the corresponding time slots as the input of a GRU, and the output is transformed to a prediction of taxi demands at the next time slot by a FC layer. The computations of the short-term periodic module are defined as follows,

$$\mathbf{f}^s = FC(GRU(\mathbf{X}_{\tau+1-p\Delta_s} \oplus \mathbf{u}_{\tau+1-p\Delta_s}, \dots, \mathbf{X}_{\tau+1-\Delta_s} \oplus \mathbf{u}_{\tau+1-\Delta_s})), \quad (3)$$

where $\oplus$ denotes the concatenation operation, and $\mathbf{f}^s = [f_1^s, f_2^s, \dots, f_N^s]$. Similarly, the prediction output by the long-term periodic module can be computed as

$$\mathbf{f}^l = FC(GRU(\mathbf{X}_{\tau+1-q\Delta_l} \oplus \mathbf{u}_{\tau+1-q\Delta_l}, \dots, \mathbf{X}_{\tau+1-\Delta_l} \oplus \mathbf{u}_{\tau+1-\Delta_l})), \quad (4)$$

where $\mathbf{f}^l = [f_1^l, f_2^l, \dots, f_N^l]$.

3.4 Context-Aware Attention Module

Three predictions (i.e., $\mathbf{f}^i$, $\mathbf{f}^s$, and $\mathbf{f}^l$) have been output by the previous modules. In this subsection, we leverage attention mechanism to incorporate the three predictions by considering context information which is our first attempt. In what follows, we first introduce how to extract context features, and then explain our context-aware attention network structure.

As shown in Fig. 3, we construct a *context feature vector* for each region r_n at time slot $t_{\tau+1}$, denoted by $\mathbf{g}_n$. Here, we consider three main context factors, including weather condition at $t_{\tau+1}$, index of time slots $t_{\tau+1}$, and function of region r_n , which make a difference to taxi demands. Specially, we use the same method in feature extraction of external factors, to embed the index of time slots into a low-dimensional vector, and concatenate it with the vectors of weather condition and region function.

Next, we construct a perceptron to learn the attention should be paid to the three predictions of each region. Figure 3 presents the detailed structure of the perceptron, which is composed of two FC layers and a softmax operation. It takes the context feature vector and taxi demand predictions of region r_n at $t_{\tau+1}$ and outputs a 1×3 vector, denoted by $\mathbf{w}_n = [w_n^i, w_n^s, w_n^l]$. The three elements of the vector can be interpreted as the weights assigned to the predictions of r_n . Thus, we can obtain the final taxi demand prediction of r_n at $t_{\tau+1}$ by computing the weighted sum of its three predictions, i.e.,

$$\hat{X}_{n,\tau+1} = w_n^i \cdot f_n^i + w_n^s \cdot f_n^s + w_n^l \cdot f_n^l. \quad (5)$$

Note that the weights indicate that to what extent the predictions should be noticed.

3.5 Learning

Since the taxi demand prediction is a regression problem, we adopt mean square error as our loss function, and train our network model by minimizing the error between prediction $\hat{\mathbf{X}}_{\tau+1}$ and ground truth $\mathbf{X}_{\tau+1}$, i.e.,

$$L_{loss}(\Omega) = \|\mathbf{X}_{\tau+1} - \hat{\mathbf{X}}_{\tau+1}\|_2^2, \quad (6)$$

where Ω is the set of all learnable parameters in our network model.

4 Experiments

4.1 Datasets

We first introduce the real-world datasets from New York City (NYC) used in our experiments. **Road Network Data.** The road network of NYC consists of 87,898 intersections and 91,649 road segments. In this paper, we partition NYC into 972 regions by road segments, as shown in Fig. 4 (The averaged number of taxi demands on weekdays are plotted, in which deeper color indicates more taxi

demands). **Taxi Tripping Data.** An open dataset of taxi tripping records in NYC [1], which contains the detailed driving information like the pick-up and drop-off locations and timestamps of each trip. The taxi tripping data from Jan. 1, 2016 to Jun. 30, 2016 (130 weekdays and 52 weekends) are used, containing 87,866,988 trips. **POI Data.** We use a POI dataset with 670,916 POIs in NYC, classified into 16 different categories. **Meteorological and Holiday Data.** A dataset of meteorological records from Jan. 1, 2016 to Jun. 30, 2016 is also used in our work, containing weather condition (e.g., sunny and rainy), temperature, wind speed, and humidity information recorded every six hours. We also consider 10 statutory holidays in the United States.

4.2 Compared Methods

We compare our proposed model with the following baselines.

- **Historical Average (HA):** predicts taxi demands at the next time slot in each region by averaging the historical taxi demands at the same time slot.
- **Autoregressive Integrated Moving Average (ARIMA):** is a widely-used method for time-series prediction problems, which model the temporal dependencies by combining moving averages and autoregressive components.
- **Long Short Term Memory (LSTM):** is a variant of recurrent neural networks, which can effectively learn underlying dependencies in long and short term from a sequence data.
- **Diffusion Convolution Recurrent Neural Network (DCRNN) [7]:** integrates graph convolution into gated recurrent units to predict traffic flows on the road network. In this model, bidirectional graph random walk operation is employed, to extract the spatial dynamics of the traffic flows, and the temporal dynamics are captured by RNN.
- **Spatial-Temporal Graph Convolutional Networks (STGCN) [13]:** consists of several ST-Conv blocks, which are built with entirely convolutional layers, to tackle traffic prediction tasks. Specifically, each block is composed of graph convolution and gated temporal convolution, which jointly process graph-structured time series.

We also analyze the performance achieved by different modules of our model, to study their effectiveness in taxi demand prediction.

- **Instant Spatial-Temporal Module (ISTM):** we only use the instant spatial-temporal module, which includes LC layers and a GRU.
- **Short/Long-Term Periodic Module and Context-Aware Attention Module (PM+CAAM):** we only use two periodic modules with considering short-term and long-term periods. The outputs of the two modules are fused by the context-aware attention module.
- **Instant Module w/o LC, Short/Long-Term Periodic Module, and Context-Aware Attention Module (IM+PM+CAAM):** we use the instant spatial-temporal module without LC layers and the periodic modules to predict taxi demands respectively. The outputs of the three modules are fused by the context-aware attention module.

- **Instant Spatial-Temporal Module and Short/Long-Term Periodic Module (ISTM+PM):** we only use the instant spatial-temporal module and the periodic modules, and their outputs are fused by a weight tensor which is learned during network training.

4.3 Default Setting

The default values of parameters in our experiments are set up as follows. We set a time slot as 15 min. In the instant spatial-temporal module, six successive time slots are used, i.e., $o = 6$. In addition, we set $k = 16$, $M = 14$, and $L = 3$ in the default setting. In the short-term and long-term periodic modules, 4 and 2 periodic time slots are employed respectively, with intervals $\Delta_s = 96$ (a day) and $\Delta_l = 96 \times 7$ (a week). We embed weather condition, holiday condition, and weekend into a 1×3 vector, respectively. The numbers of hidden nodes in GRUs in the three modules are all set as 512. In the context-aware attention module, we embed index of time slots into a 1×5 vector.

We use the historical records during Jun. 2016 as testing data, and the rest records as training data. The performance achieved by each method is evaluated by root mean square error (RMSE) and mean absolute error (MAE). Besides, we adopt Adam optimization algorithm for training parameters. The learning rate of Adam is set as 10^{-4} , and the batch size during training is 64. We also employ early stop in our experiments, in which the number of rounds and the maximal epoch are set as 6 and 100, respectively. All experiments are conducted on a NVIDIA RTX2070 graphics card, and experimental results are the average of five runs under the same setting with different random seeds.

4.4 Experimental Results

Comparison with Baselines. Table 2 shows the performance achieved by our proposed model and the baselines under the default setting. We can easily find that our model achieves the lowest RMSE (3.209) and MAE (1.119), compared with all the baselines. Specifically, HA and ARIMA perform the poorest, which achieves 81.8% (46.1%) and 196.9% (145.0%) higher RMSE (MAE) than our proposed model, respectively. It demonstrates that deep neural networks (e.g., LSTM) can work effectively in urban data prediction. Furthermore, LSTM achieves worse performance than our model, as it only models the temporal dependencies in taxi demands. In the baselines, STGCN and DCRNN achieve good performance, which capture both spatial and temporal correlations. Compared with STGCN and DCRNN, our model achieves 8.4% (10.8%) and 9.0% (11.8%) lower RMSE (MAE), respectively.

Evaluation of Modules. We also evaluate the effectiveness of different modules in our model, which is shown in Table 3. It can be easily found that each module in our model works in terms of achieving better prediction performance. Specially, by comparing the results of ISTM and ISTM+PM, we confirm that

the periodic modules work. As PM+CAAC achieves worse performance than our model, we can know that the instant spatial-temporal module is useful. The effectiveness of LC layers, which extract spatial correlations of taxi demands in different regions, is verified by comparing the results of IM+PM+CAAM and our model. Moreover, our model achieves better performance than ISTM+PM, which confirms the usefulness of the context-aware attention module.

5 Related Work

Model-Based Methods. Some model-based prediction methods are provided in the earlier works [6,8,9,15], to capture the intrinsic patterns of historical taxi demands. For example, Li et al. [6] model taxi demand prediction as a time series prediction problem, and an improved ARIMA method is developed to predict taxi demands by leveraging the temporal dependencies. Tong et al. [9] propose a unified linear regression model with high-dimensional features to predict taxi demands for each region. Due to a lack of nonlinear modeling capabilities, these methods usually have low prediction accuracy.

Fig. 4. Regions

Table 2. Performance comparison

Methods	RMSE	MAE
HA	5.835	1.635
ARIMA	9.530	2.742
LSTM	3.650	1.286
DCRNN	3.497	1.251
STGCN	3.479	1.240
CACRNN	3.209	1.119

Table 3. Evaluation of different components

Methods	RMSE	MAE
ISTM	3.464	1.280
PM+CAAM	3.974	1.302
IM+PM+CAAM	3.485	1.186
ISTM+PM	3.354	1.213
CACRNN	3.209	1.119

DNN-Based Methods. Recently, deep neural networks, such as convolutional neural networks (CNN) and recurrent neural networks (RNN), are widely used in taxi demand prediction, to capture spatial and temporal features. Fully-connected layers and residual networks are employed in [10] to automatically learn features to assist taxi demand prediction. Xu et al. [11] propose a LSTM-based sequential learning framework to model temporal dependencies of taxi demand in recent moments. Furthermore, Yao et al. [12] adopt CNN and LSTM to extract the spatial correlations among adjacent regions and temporal dependencies in a close period, respectively. External factors are further leveraged in [2] to improve the prediction accuracy. Chu et al. [4] try to incorporate the spatial-temporal dependencies and external factors by using fixed parameter matrixes learned during model training. However, all the above existing works are limited

in incorporating different spatial-temporal features and external factors together, since fixed notice is paid to them without considering the impacts of contextual information.

6 Conclusion

In this paper, we propose a context-aware attention-based convolutional recurrent neural network to predict fine-grained taxi demands. We capture multi-view features, i.e., spatial correlations among adjacent regions, short-term periodicity, long-term periodicity, and impacts of external factors, by adopting the LC layers and GRUs. More important, we develop a context-aware attention network to incorporate the predictions of each region, by assigning them different weights according to contextual information. The weights indicate that to what extent the predictions should be noticed. Finally, comprehensive experiments are conducted based on real-world multi-source datasets. The results show that our method achieves 8.4% (10.8%) and 9.0% (11.8%) improvement in RMSE (MAE) over two state-of-the-art methods, STGCN and DCRNN.

Acknowledgment. This research is supported by National Natural Science Foundation of China (NSFC) under Grant No. 61802245 and the Shanghai Sailing Program under Grant No. 18YF1408200. This work is also supported by Science and Technology Commission Shanghai Municipality (STCSM) (No. 19511121002).

References

1. New York taxi dataset. <https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page>
2. Bai, L., Yao, L., Kanhere, S.S., Yang, Z., Chu, J., Wang, X.: Passenger demand forecasting with multi-task convolutional recurrent neural networks. In: Yang, Q., Zhou, Z.-H., Gong, Z., Zhang, M.-L., Huang, S.-J. (eds.) PAKDD 2019. LNCS (LNAI), vol. 11440, pp. 29–42. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-16145-3_3
3. Cho, K., et al.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1724–1734 (2014)
4. Chu, J., et al.: Passenger demand prediction with cellular footprints. In: 2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), pp. 1–9. IEEE (2018)
5. Gal, Y., Ghahramani, Z.: A theoretically grounded application of dropout in recurrent neural networks. In: Advances in Neural Information Processing Systems, pp. 1019–1027 (2016)
6. Li, X., et al.: Prediction of urban human mobility using large-scale taxi traces and its applications. *Frontiers Comput. Sci.* **6**(1), 111–121 (2012)
7. Li, Y., Yu, R., Shahabi, C., Liu, Y.: Diffusion convolutional recurrent neural network: data-driven traffic forecasting. In: International Conference on Learning Representations, ICLR 2018 (2018)

8. Moreira-Matias, L., Gama, J., Ferreira, M., Damas, L.: A predictive model for the passenger demand on a taxi network. In: 2012 15th International IEEE Conference on Intelligent Transportation Systems, pp. 1014–1019. IEEE (2012)
9. Tong, Y., et al.: 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, pp. 1653–1662. ACM (2017)
10. Wang, D., Cao, W., Li, J., Ye, J.: DeepSD: supply-demand prediction for online car-hailing services using deep neural networks. In: 2017 IEEE 33rd International Conference on Data Engineering (ICDE), pp. 243–254. IEEE (2017)
11. Xu, J., Rahmatizadeh, R., Bölöni, L., Turgut, D.: Real-time prediction of taxi demand using recurrent neural networks. *IEEE Trans. Intell. Transp. Syst.* **19**(8), 2572–2581 (2017)
12. Yao, H., et al.: Deep multi-view spatial-temporal network for taxi demand prediction. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
13. Yu, B., Yin, H., Zhu, Z.: Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence, pp. 3634–3640. AAAI Press (2018)
14. Yuan, N.J., Zheng, Y., Xie, X.: Segmentation of urban areas using road networks. MSR-TR-2012-65, Technical report (2012)
15. Zhang, L., Chen, C., Wang, Y., Guan, X.: Exploiting taxi demand hotspots based on vehicular big data analytics. In: 2016 IEEE 84th Vehicular Technology Conference, VTC-Fall, pp. 1–5. IEEE (2016)