Transportation Research Part C 85 (2017) 591–608

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach

Jintao Ke[a], Hongyu Zheng[b], Hai Yang[a], Xiqun (Michael) Chen[b,*]

[a] Department of Civil and Environmental Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong, China
[b] College of Civil Engineering and Architecture, Zhejiang University, Hangzhou, China

ARTICLE INFO

Keywords:
On-demand ride services
Short-term demand forecasting
Deep learning (DL)
Fusion convolutional long short-term memory network (FCL-Net)
Long short-term memory (LSTM)
Convolutional neural network (CNN)

ABSTRACT

Short-term passenger demand forecasting is of great importance to the on-demand ride service platform, which can incentivize vacant cars moving from over-supply regions to over-demand regions. The spatial dependencies, temporal dependencies, and exogenous dependencies need to be considered simultaneously, however, which makes short-term passenger demand forecasting challenging. We propose a novel deep learning (DL) approach, named the fusion convolutional long short-term memory network (FCL-Net), to address these three dependencies within one end-to-end learning architecture. The model is stacked and fused by multiple convolutional long short-term memory (LSTM) layers, standard LSTM layers, and convolutional layers. The fusion of convolutional techniques and the LSTM network enables the proposed DL approach to better capture the spatio-temporal characteristics and correlations of explanatory variables. A tailored spatially aggregated random forest is employed to rank the importance of the explanatory variables. The ranking is then used for feature selection. The proposed DL approach is applied to the short-term forecasting of passenger demand under an on-demand ride service platform in Hangzhou, China. The experimental results, validated on the real-world data provided by DiDi Chuxing, show that the FCL-Net achieves the better predictive performance than traditional approaches including both classical time-series prediction models and state-of-art machine learning algorithms (e.g., artificial neural network, XGBoost, LSTM and CNN). Furthermore, the consideration of exogenous variables in addition to the passenger demand itself, such as the travel time rate, time-of-day, day-of-week, and weather conditions, is proven to be promising, since they reduce the root mean squared error (RMSE) by 48.3%. It is also interesting to find that the feature selection reduces 24.4% in the training time and leads to only the 1.8% loss in the forecasting accuracy measured by RMSE in the proposed model. This paper is one of the first DL studies to forecast the short-term passenger demand of an on-demand ride service platform by examining the spatio-temporal correlations.

1. Introduction

The on-demand ride service platform, e.g., Urber, Lyft, DiDi Chuxing, is an emerging technology with the boom of the mobile internet. Ride-sourcing or transportation network companies (TNCs) refer to an emerging urban mobility service mode that private car owners drive their own vehicles to provide for-hire rides (Chen et al., 2017). On-demand ride-sourcing services can be completed via smart phone applications. The platform serves as a coordinator who matches requesting orders from passengers (demand) and

[*] Corresponding author.
E-mail address: chenxiqun@zju.edu.cn (X.M. Chen).

http://dx.doi.org/10.1016/j.trc.2017.10.016
Received 20 June 2017; Received in revised form 15 October 2017; Accepted 15 October 2017

vacant registered cars (supply). There exists an abundance of leverages to influence drivers' and passengers' preference and behavior, and thus affect both the demand and supply, to maximize profits of the platform or achieve the maximum social welfare. Having a better understanding of the short-term passenger demand over different spatial zones is of great importance to the platform or the operator, who can incentivize drivers to the zones with more potential passenger demands, and improve the utilization rate of the registered cars.

Although limited research efforts have been implemented on forecasting short-term passenger demand under the emerging on-demand ride service platform in most recent years mainly due to the real-world data unavailability, the fruitful studies on the taxi market can provide valuable insights since there exist strong similarities between the taxi market and the on-demand ride service market. A series of mathematical models were developed to spell out endogenous relationships among variables in the taxi market (; Yang et al., 2002, 2005, 2010b; Yang and Yang, 2011) under the two-sided market equilibrium. On the demand side, the accurate passenger demand was affected by passengers' waiting time and taxi fare; while on the supply side, drivers' behavior, i.e., how to find a passenger, was mainly affected by the expected searching time and taxi fare. The passenger demand was endogenously determined when the taxi operator decided the taxi fare structure and the number of released licenses of taxis (entry limitation).

In theory, the equilibrium between the demand and supply will eventually be reached when the arrival rate of passengers equals to the arrival rate of vacant taxis and equals to the meeting rate. However, heterogeneous and exogenous factors in reality, e.g., the asymmetric information, and short-term fluctuations, may make it difficult to guarantee the spatial distribution of taxis matching the passenger demand all the time (Moreira-Matias et al., 2013). Hence, disequilibrium states can result from the following two scenarios: oversupply (an excess in the number of vacant taxis may decrease the taxi utilization) and overfull demand (excessively waiting passengers may lower the degree of satisfaction). Both scenarios are harmful to the taxi operator as well as the on-demand ride service platform, raising a strong need for a precise forecasting of short-term passenger demand. It helps the operator/platform implement proactive incentive mechanism, such as surge pricing and cash/point awards, to attract drivers from regions of oversupply to regions with overfull demand. These strategies not only shorten the process of reaching equilibrium under a dynamic environment but also help improve the taxi/car utilization rate and reduce passengers' waiting time.

However, short-term forecasting of passenger demand or on-demand ride services in each region is of great challenge mainly due to the three kinds of dependencies (Zhang et al., 2017b):

(1) Time dependencies: the passenger demand has a strong periodicity (for example, the passenger demand is expected to be high during morning and evening peaks and to be low during sleeping hours); furthermore, the short-term passenger demand is dependent on the trend of the nearest historical passenger demand.
(2) Spatial dependencies: Yang et al. (2010b) revealed that the passenger demand in one specific zone was not merely determined by the variables of this zone, but endogenously dependent on all the zonal variables in the whole network. Generally, the variables of the nearby zones have stronger influences than distant zones, which inspires the need for an advanced model that can capture local spatial dependencies.
(3) Exogenous dependencies: some exogenous variables, such as the travel time rate and weather conditions, may have strong influences on the short-term passenger demand. The exogenous variables also demonstrate time dependencies and spatial dependencies.

Although little direct experience suggests solutions to these three dependencies in short-term passenger demand forecasting, studies on traffic speed/volume prediction and rainfall nowcasting provide valuable insights (Ghosh et al., 2009; Huang and Sadek, 2009; Guo et al., 2014; Wang et al., 2014). Recently, deep learning (DL) approaches have been successfully used for traffic flow prediction. For example, Ma et al. (2015a) employed the long short-term memory (LSTM) neural network to capture the long-term dependencies and nonlinear traffic dynamics for short-term traffic speed prediction. Zhang et al. (2017b) presented a deep spatio-temporal residual network to predict the inflow and outflow in each region of a city simultaneously. Shi et al. (2015) innovatively integrated CNN and LSTM in one end-to-end DL structure, named the convolutional LSTM (conv-LSTM), which provided a brand-new idea for solving spatio-temporal sequence forecasting problems. In that research, the numerical experiments showed that the conv-LSTM outperformed fully connected LSTM in two datasets.

In this paper, we propose a novel DL structure, named the fusion convolutional LSTM network (FCL-Net), to consider the three dependencies simultaneously in the short-term passenger demand forecasting for the on-demand ride service platform. Different from the aforementioned studies, this structure coordinates the spatio-temporal variables and non-spatial time-series variables in one end-to-end trainable model. Before feeding these explanatory variables into the DL structure, a tailored spatial aggregated random forest is designed to evaluate the feature importance with different categories, look-back time intervals, and spatial locations.

To the best knowledge of the authors, this paper is one of the first attempts to employ spatio-temporal DL approaches in short-term passenger demand forecasting under the on-demand ride service platform. The main contributions of this paper are within three folds:

(1) The novel FCL-Net approach characterizes the spatio-temporal properties of the spatio-temporal predictors, captures the temporal features of non-spatial time-series variables simultaneously, and coordinates them in one end-to-end learning structure for the short-term passenger demand forecasting.
(2) We propose a spatial aggregated random forest to extract the potential predictors affecting short-term passenger demand and assess the feature importance of them.
(3) Validated by the real-world on-demand ride services data provided by DiDi Chuxing in a large-scale urban network, the proposed

DL structure outperforms both the traditional and state-of-the-art benchmark algorithms, including three conventional time-series models and several efficient machine learning/deep learning approaches.

The rest of the paper is organized as follows. Section 2 first reviews the existing research on the taxi market modeling and taxi-passenger demand forecasting, and then summarizes the state-of-the-art DL approaches utilized in related problems. Section 3 formulates the short-term traffic flow forecasting problem, and explicitly explains the explanatory variables. Section 4 describes the structure and mathematical formulation of the proposed FCL-Net, as well as the proposed spatial aggregated random forest algorithm for feature selection. Section 5 compares the predictive performance between the proposed approach and the benchmark models based on the real-world dataset extracted from DiDi Chuxing. Finally, Section 6 concludes the paper and outlooks future research.

2. Literature review

The fast-growing technology of mobile internet enables on-demand ride service platforms for providing efficient connections between waiting passengers and vacant registered cars. Like the taxi market, the on-demand ride service market is essentially a two-sided market where both the consumers (passengers) and providers (drivers of vacant cars) are independent and have individual mode choices. The passengers make mode choice decisions between taxi/on-demand ride service and public transportation according to the waiting time and trip fare, while the drivers make service decisions by considering the searching time and trip fare. In light of the fact that the vacant taxis and waiting passengers are unable to be matched simultaneously in a specific zone, Yang et al. (2002, 2010b) and Yang and Yang (2011) proposed a meeting function to characterize the search frictions between drivers of vacant taxis and waiting passengers. The meeting function pointed out that the meeting rate in one specific zone was determined by the density of the waiting passengers and vacant taxis at that moment, which indicated that passengers' waiting time, drivers' searching time, and passengers' arrival rate (demand) were endogenously correlated. The equilibrium state was reached when the arrival rate of waiting passengers exactly matched the arrival rate of vacant taxis. This equilibrium state along with the endogenous variables were influenced by the exogenous variables, such as the taxi fleet size and taxi trip fare. The taxi operator might coordinate supply and demand via the on-demand service platform and thus influence the equilibrium state by regulating the entry of taxi and determining the taxi fare structure, such as non-linear pricing (Yang et al., 2010a). Apart from the traditional taxi market, some emerging market structures, like the ride-sourcing market (Zha et al., 2016), e-hailing taxi market (He and Shen, 2015; X. Wang et al., 2016), were examined under the same equilibrium modeling framework. Recently, the on-demand ride-sharing market and the optimal assignment strategies have also attracted researchers' attentions(Alonso-Mora et al., 2017).

However, researchers found that a regional disequilibrium occurred when there was an excess in vacant taxis or waiting passengers in that region (Moreira-Matias et al., 2012). This disequilibrium might lead to a resource mismatch between supply and demand, which resulted in the low taxi utilization in some regions while low taxi availability in other regions. Therefore, a short-term passenger demand forecasting model is of great importance to the taxi operator, which can implement efficient taxi dispatching and time-saving route finding to achieve an equilibrium across urban regions (Zhang et al., 2017a). To attain the accurate and robust short-term passenger demand forecasting, both parametric (e.g., ARIMA) and non-parametric models (e.g., neural network) have been examined. For instance, Zhao et al. (2016) implemented and compared three models, i.e., the Markov algorithm, Lempel-Ziv-Welch algorithm, and neural network. In that research, the results showed that neural network performed better with the lower theoretical maximum predictability while the Markov predictor had better performance with the higher theoretical maximum predictability. Moreira-Matias et al. (2013) proposed a data stream ensemble framework which incorporated time varying passion model and ARIMA, to predict the spatial distribution of taxi passenger demand. Deng and Ji (2011) employed the global and local Moran's I values to evaluate the intensity of taxi services in Shanghai. Some socio-demographical and built-environment variables have also been in use for predicting taxi passenger demand (Qian and Ukkusuri, 2015).

There are a broad range of problems in the domain of transportation, which are similar to short-term passenger demand forecasting. These problems include the traffic speed estimation (Bachmann et al., 2013; Soriguera and Robusté, 2011; Wang and Shi, 2013), traffic volume prediction (Boto-Giralda et al., 2010), real-time crash likelihood estimation (Ahmed and Abdel-Aty, 2013; Yu et al., 2014), car-following behavior prediction(Wang et al., 2017), original-destination matrices forecasting (Toqué et al., 2016), bus arrival time prediction (Yu et al., 2011), short-term forecasting of high speed rail demand (Jiang et al., 2014), and etc., the solutions to which offer meritorious inspirations to our problem. To solve these spatio-temporal forecasting problems, a broad range of approaches have been proposed, including the ARIMA family (Zhang et al., 2011; Khashei et al., 2012), local regression model (Antoniou et al., 2013), neural network based algorithms (Chan et al., 2012), and Bayesian inferring approaches (Fei et al., 2011). Vlahogianni et al. (2014) reviewed the existing literature on short-term traffic forecasting, and observed that researchers were moving from classical statistical models to neural network based approaches with the explosive growth of data accessibility and computing power.

Recently, more and more DL algorithms have been utilized in traffic prediction due to their capability of capturing complex relationship from a huge amount of data. Cheng et al. (2016) proposed a DL based approach to forecast day-to-day travel demand variations in a large-scale traffic network. Huang et al. (2014) predicted short-term traffic flow via a two-layer DL structure with a deep belief network (DBN) at the bottom and a multitask regression model (MTL) at the top. Polson and Sokolov (2017) found that the sharp nonlinearities of traffic flow, as a result of transitions between the free flow, breakdown, recovery and congestion, could be captured by a DL architecture. Combining the empirical mode decomposition (EMD) and back-propagation neural network (BPN), Wei and Chen (2012) presented a hybrid EMD-BPN method for short-term passenger flow forecasting. Graphical LASSO was also combined in the neural network, showing its potential in network-scale traffic flow forecasting (Sun et al., 2012). Lv et al. (2015)

stated that a stacked autoencoder model helped to capture generic traffic flow features and characterize spatial temporal correlations in traffic flow prediction. Ma et al. (2015b) extended the deep learning theory for the large-scale traffic network analysis, and predicted the evolution of traffic congestion with the help of taxi GPS data.

One of the obstacles in traffic forecasting is how to capture spatio-temporal correlations. It was found that the vehicle accumulation and dissipation had impacts on the travel volume of adjacent links or intersections, which indicated the spatial correlations should be considered in forecasting (Zhu et al., 2014). In terms of the spatial correlations, CNN developed by (LeCun et al., 1999) was used to learn the local and global spatial correlations in large-scale, network-wide traffic forecasting (Chen et al., 2016). To address temporal correlations (another inherit property in real-time traffic forecasting), the family of recurrent neural networks (RNN) (Williams and Zipser, 1989) was widely viewed as one of the most suitable structures (Zhao et al., 2017). In the RNN architecture, the dependent variable in one timestamp was not only dependent on the explanatory variables in this timestamp, but also correlated with the explanatory variables in the previous timestamps (Sutskever et al., 2009). However, the traditional RNN suffered from a "vanishing gradience" effect which made it impossible to store long-term information (Hochreiter, 1991). To address this issue, Hochreiter and Schmidhuber (1997) presented the long short-term memory (LSTM) which employed a series of memory cells to store information for exploring long-range dependencies in the data.

However, neither CNN nor LSTM are perfect models for spatio-temporal forecasting problems. CNN fails to capture the temporal dependencies while LSTM is incapable of characterizing local spatial correlations. To capture spatial and temporal dependencies simultaneously in one end-to-end training model, researchers have made numerous attempts in recent years. J. Wang et al. (2016) proposed a novel error-feedback recurrent convolutional neural network (eRCNN) architecture which was comprised of the input layer, the convolutional layer, and the error-feedback recurrent layer. Zhang et al. (2017b) modeled the temporal closeness, period, and trend properties of the inflow/outflow of human mobilities with serval separate convolutional layers and then fused these layers in one end-to-end DL structure. Ma et al. (2017) proposed a deep convolutional neural network for large-scale traffic network speed prediction, where the space-time matrix was converted to an image as the input of the CNN. Yu et al. (2017) designed a spatio-temporal recurrent convolutional network for predicting network-wide traffic speeds. This model was comprised of two components: the lower one modeled the spatial features while the upper one learned the temporal features. Shi et al. (2015) proposed a conv-LSTM network, which combined CNN and LSTM in one sequence to sequence learning framework, for precipitation nowcasting that was a typical spatio-temporal forecasting problem. This network structure was different from the previous attempts to combine CNN and LSTM, as it redefined the mathematical formulations of LSTM by transferring the vector-based LSTM cell to a tensor-based LSTM cell (which enabled convolutional operations), instead of merely stacking CNN and LSTM. In that research, the results showed that the conv-LSTM outperformed the fully-connected LSTM, since some complicated spatio-temporal characteristics could be learned by the convolutional and recurrent structure of the model.

However, short-term passenger demand is not only dependent on its own spatio-temporal properties but also dependent on other explanatory variables (some with spatio-temporal properties and some only with temporal properties). Therefore, we extend the conv-LSTM to a more generalized framework which is comprised of two categories of structures, one of which characterizes spatio-temporal features while the other captures non-spatial temporal features. In this way, the aforementioned three dependencies (spatial, temporal, and exogenous) can be addressed at the same time.

3. Preliminaries

The short-term passenger demand forecasting is essentially a time-series prediction problem, which implies that the nearest historical passenger demand can provide valuable information for predicting the future demand. We also observe that the travel time rate influences the short-term passenger demand, since it reflects the congestion level of trips and zones. For example, passengers will potentially transfer to the metro transit if they find the trips to their destinations are congested. Furthermore, the attributes of time-of-day, day-of-week, and weather conditions also have impacts on the short-term passenger demand.

In this section, we first interpret the notations of the variables used in this paper, then give an explicit definition of the short-term passenger demand forecasting problem.

Definition 1 (*Region and time partition*). The urban area is partitioned into $I \times J$ grids uniformly where each grid refers to a zone. On the other hand, we consider variables aggregated in a one-hour time interval in this paper.Based on Definition 1, we explicitly define several categories of variables as follows:

(1) Demand intensity

The intensity of demand at the tth time slot (e.g., hour) lying in grid (i,j) is defined as the number of orders during this time interval within the grid, which is denoted by $d_t^{i,j}$. The intensity of demand in all $I \times J$ grids at the tth time slot is defined as the matrix $\boldsymbol{D}_t \in R^{I\times J}$ (R refers to the real set), where the (i,j)th element is $(\boldsymbol{D}_t)_{i,j} = d_t^{i,j}$.

(2) Average travel time rate

The travel time rate represents the travel time per unit travel distance (Chen et al., 2017). In this paper, the travel time rate of the mth order originating from grid (i,j) at the tth time slot, is defined as the ratio of its travel time to its travel distance, $\tau_{t,m}^{i,j}$. The average travel time rate in grid (i,j) during the tth time slot, $\tau_t^{i,j}$, is defined as the average of $\tau_{t,m}^{i,j}$ over m. The average travel time rate in all $I \times J$ grids at the tth time slot is defined as the matrix $\Gamma_t \in R^{I\times J}$, where the (i,j)th element is $(\Gamma_t)_{i,j} = \tau_t^{i,j}$.

(3) Time-of-day and day-of-week

By empirically examining the distribution of demand intensity with respect to time in the training set, 24 h in each day can be intuitively divided into 3 periods: peak hours, off-peak hours, and sleep hours. We simply rank the hours based on the empirical demand intensity, and define the top 8 h, middle 8 h, bottom 8 h, as the peak hours, off-peak hours and sleep hours. We further introduce the dummy variable h_t to characterize this attribute of time-of-day, given by

$$h_t = \begin{cases} 2, & \text{if } t \text{ belongs to peak hours} \\ 1, & \text{if } t \text{ belongs to off-peak hours} \\ 0, & \text{if } t \text{ belongs to sleep hours} \end{cases}$$

We also denote another dummy variable w_t to be the day-of-week, which catches up the distinguished properties between weekdays and weekends.

$$w_t = \begin{cases} 0, & \text{if } t \text{ belongs to weekdays} \\ 1, & \text{if } t \text{ belongs to weekends} \end{cases}$$

(4) Weather

We consider 5 categories of weather variables, including temperature (measured by Celsius degree), relative humidity (measured by percentage), weather state, wind speed (measured by mile per hour), and visibility (measured by kilometers). The weather state variable contains 5 categories: sunny (5), cloudy (4), light rain (3), moderate rain (2), and heavy rain (1). The weather state takes one value for one hour, while temperature, humidity, weather state, wind speed, and visibility are taken average in each hour. In this paper, the temperature, humidity, weather state, wind speed, and visibility during the tth time interval are denoted as at_t,ah_t,as_t,aw_t,av_t, respectively.

All of the aforementioned variables demonstrate the time-varying attributes, but the demand intensity and average travel time rate show the zonal-based attributes, which mean they have different values across grids. The variables with time-varying attributes only have temporal dependencies, while the variables with time-varying and zonal-based attributes have both spatial and temporal dependencies, which implies that they should be treated in different ways. Thus, we give the definition of the spatio-temporal variables and non-spatial time-series variables in Definition 2.

Definition 2 (*Spatio-temporal variables*). Refer to the variables showing distinction across time and across space, which imply there exist spatio-temporal correlations, e.g., the demand intensity, and travel time rate. Other variables, including the time-of-day, day-of-week, and weather variables, are denoted as non-spatial time-series variables, which vary across time instead of space.With the aforementioned definition of the explanatory variables, we can formulate the short-term passenger demand forecasting as Problem 1.

Problem 1. Given the historical observations and pre-known information $\{\boldsymbol{D}_s,\boldsymbol{\Gamma}_s|s = 0,\ldots,t-1;h_s,w_s|s = 0,\ldots,t;at_s,ah_s,as_s,aw_s,av_s|s = 0,\ldots,t-1\}$, predict $\boldsymbol{D}_t$. It is noteworthy that the time-of-day and day-of-week of the tth time slot (h_t and w_t) are pre-known at t.

4. Methodology

In this paper, we propose a novel DL architecture, i.e., FCL-Net, to capture the spatial dependencies, temporal dependencies, and exogenous dependencies, in short-term passenger demand forecasting. To reduce the computation complexity, we also present a spatial aggregated random forest algorithm to rank the importance of explanatory variables and select the important ones. In this section, we first present a brief review of the traditional LSTM and conv-LSTM, then introduce the proposed architecture and training algorithm of FCL-Net, and finally illustrates the proposed spatial aggregated random forest.

4.1. LSTM and Conv-LSTM

The traditional artificial neural network (ANN) lacks the ability to catch up time-series characteristics since it does not take the temporal dependencies into consideration. To overcome this shortcoming, RNN is proposed, where the connection between units is organized by timestamps. The inner structure of an RNN layer is illustrated in Fig. 1, where the input is a T time-stamp vector sequence $\boldsymbol{x} = (\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_T)$ and the output is a hidden vector sequence $\boldsymbol{h} = (\boldsymbol{h}_1,\boldsymbol{h}_2,\ldots,\boldsymbol{h}_T)$. It is noteworthy that $\boldsymbol{x}_t$ can be a one-dimensional vector or scalar, while $\boldsymbol{h}_t$ does not necessarily have the same dimension as $\boldsymbol{x}_t$. The hidden unit value in timestamp t, i.e., $\boldsymbol{h}_t$,

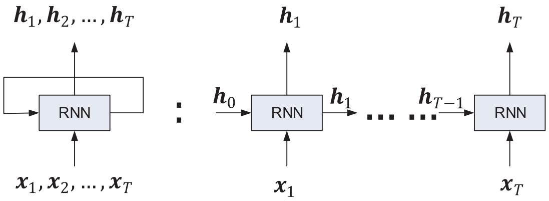


Fig. 1. Illustration of the inner structure of an RNN layer.

stores the information, including hidden values ($\boldsymbol{h}_1,\boldsymbol{h}_2,...,\boldsymbol{h}_{t-1}$) and input values ($\boldsymbol{x}_1,\boldsymbol{x}_2,...,\boldsymbol{x}_{t-1}$), of the previous timestamps. Together with the input in t, i.e., $\boldsymbol{x}_t$, it is passed to the next timestamp $t + 1$ at each iteration. In this way, RNN can memorize the information from multiple previous timestamps. Although RNN exhibits strong ability in catching temporal characteristics, it fails to store information for a long-term memory.

LSTM, as a special RNN structure, overcomes RNN's weakness on the long-term memory. Like the standard RNN, each LSTM cell maps the input vector sequence $\boldsymbol{x}$ to a hidden vector sequence $\boldsymbol{h}$ by T iterations. As demonstrated in Eqs. (1)–(5), $\boldsymbol{i}_t,\boldsymbol{f}_t,\boldsymbol{o}_t,\boldsymbol{c}_t,(t = 1,2,...,T)$ represent the input gate, forget gate, output gate, and memory cell vectors, respectively, sharing the same dimension with $\boldsymbol{h}_t$.

$$\boldsymbol{i}_t = \sigma(\boldsymbol{W}_{xi}\boldsymbol{x}_t + \boldsymbol{W}_{hi}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{ci}\circ\boldsymbol{c}_{t-1} + \boldsymbol{b}_i) \tag{1}$$

$$\boldsymbol{f}_t = \sigma(\boldsymbol{W}_{xf}\boldsymbol{x}_t + \boldsymbol{W}_{hf}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{cf}\circ\boldsymbol{c}_{t-1} + \boldsymbol{b}_f) \tag{2}$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t\circ\boldsymbol{c}_{t-1} + \boldsymbol{i}_t\circ\tanh(\boldsymbol{W}_{xc}\boldsymbol{x}_t + \boldsymbol{W}_{hc}\boldsymbol{h}_{t-1} + \boldsymbol{b}_c) \tag{3}$$

$$\boldsymbol{o}_t = \sigma(\boldsymbol{W}_{xo}\boldsymbol{x}_t + \boldsymbol{W}_{ho}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{co}\circ\boldsymbol{c}_t + \boldsymbol{b}_o) \tag{4}$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t\circ\tanh(\boldsymbol{c}_t) \tag{5}$$

The operator '∘' refers to Hadamard product, which calculates the element-wise products of two vectors, matrices, or tensors with the same dimensions. σ and tanh are the two non-linear activation functions given by

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{6}$$

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{7}$$

$\boldsymbol{W}_{cf},\boldsymbol{W}_{ci},\boldsymbol{W}_{co},\boldsymbol{W}_{xi},\boldsymbol{W}_{hi},\boldsymbol{W}_{xf},\boldsymbol{W}_{hf},\boldsymbol{W}_{xc},\boldsymbol{W}_{hc},\boldsymbol{W}_{xo},\boldsymbol{W}_{ho}$ are the weighted parameter matrices which conduct a linear transformation from the vector of the first subscript to the second subscript, while $\boldsymbol{b}_i,\boldsymbol{b}_f,\boldsymbol{b}_c,\boldsymbol{b}_o$ are the intercept parameters.

Multiple LSTM cells can be stacked to form a deeper and more complicated neural network, which can better discover the complex relationships between the inputs and outputs. In this paper, each LSTM cell is denoted as a function $\mathscr{F}^L$: $R^{T\times L} \to R^{T\times L'}$, where T is the length of time sequences, L is the length of one input vector, and L' is the length of one output vector (see Fig. 2).

However, LSTM is not an ideal model for the passenger demand forecasting with spatial and temporal characteristics in this paper, because it fails to capture the spatial dependencies. To overcome this shortcoming, the conv-LSTM network, which combines CNN and LSTM in one end-to-end DL architecture, is proposed.

The core idea of conv-LSTM is to transform all the inputs, memory cell values, hidden states, and various gates in Eqs. (1)–(5) to 3D tensors (shown in Eqs. (8)–(12)).

$$\mathscr{I}_t = \sigma(\boldsymbol{W}_{xi}*\mathscr{X}_t + \boldsymbol{W}_{hi}*\mathscr{H}_{t-1} + \boldsymbol{W}_{ci}\circ\mathscr{C}_{t-1} + \boldsymbol{b}_i) \tag{8}$$

$$\mathscr{F}_t = \sigma(\boldsymbol{W}_{xf}*\mathscr{X}_t + \boldsymbol{W}_{hf}*\mathscr{H}_{t-1} + \boldsymbol{W}_{cf}\circ\mathscr{C}_{t-1} + \boldsymbol{b}_f) \tag{9}$$

$$\mathscr{C}_t = \mathscr{F}_t\circ\mathscr{C}_{t-1} + \mathscr{I}_t\circ\tanh(\boldsymbol{W}_{xc}*\mathscr{X}_t + \boldsymbol{W}_{hc}*\mathscr{H}_{t-1} + \boldsymbol{b}_c) \tag{10}$$

$$\mathscr{O}_t = \sigma(\boldsymbol{W}_{xo}*\mathscr{X}_t + \boldsymbol{W}_{ho}*\mathscr{H}_{t-1} + \boldsymbol{W}_{co}\circ\mathscr{C}_t + \boldsymbol{b}_o) \tag{11}$$

$$\mathscr{H}_t = \mathscr{O}_t\circ\tanh(\mathscr{C}_t) \tag{12}$$

The input tensors, hidden tensors, memory cell tensors, input gate tensors, output gate tensors, and forget gate tensors are denoted

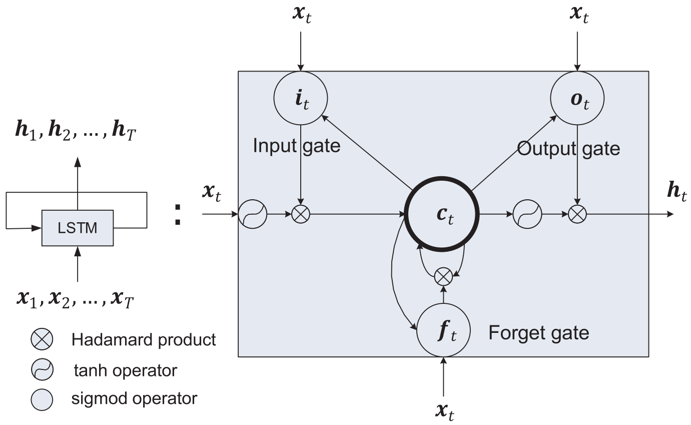


Fig. 2. Illustration of the inner structure of an LSTM layer.

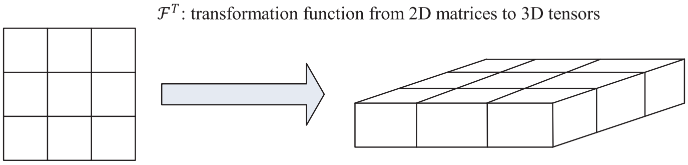

$\mathcal{F}^T$: transformation function from 2D matrices to 3D tensors

Fig. 3. Illustration of the transformation function.

as $\mathscr{X}_t,\mathscr{H}_t,\mathscr{C}_t,\mathscr{I}_t,\mathscr{O}_t,\mathscr{F}_t \in R^{M\times N\times L}$, respectively, where M,N are spatial dimensions (M rows and N columns of the grids), and L refers to the length of one input feature vector in each grid. The operator '*' stands for the convolutional operator. Here, $\boldsymbol{W}_{xf},\boldsymbol{W}_{hf},\boldsymbol{W}_{xc},\boldsymbol{W}_{hc},\boldsymbol{W}_{xo},\boldsymbol{W}_{ho}$ serve as convolutional flitters, which are replicated across the tensors with shared weights, and thus explore spatially local correlations. $\boldsymbol{b}_i,\boldsymbol{b}_f,\boldsymbol{b}_c,\boldsymbol{b}_o$ are tensors of intercept parameters, the dimension of which is consistent with the left-hand-side (for example, the dimension of $\boldsymbol{b}_i$ equals to that of $\mathscr{I}_t$). The boundary grids of the map may not have neighbor grids for convolutional operation in some directions, thus we use zero padding techniques which creates some virtual neighbor grids and fills them with zero value. In this way, the spatial dimensions (rows and columns) of the input and output of a convolutional unit are consistent.

Through these T iterations, each conv-LSTM layer can map a sequence of input tensors $\mathscr{X} = (\mathscr{X}_1,\mathscr{X}_2,...,\mathscr{X}_T)$ to a sequence of hidden tensors $\mathscr{H} = (\mathscr{H}_1,\mathscr{H}_2,...,\mathscr{H}_T)$. In this paper, each conv-LSTM cell is denoted as a function $\mathscr{F}^{CL}$: $R^{T\times M\times N\times L} \to R^{T\times M\times N\times L'}$, where T is the length of time sequences, M,N refer to dimensions of rows and columns, L' represents the length of an output feature vector in each grid respectively. Similar to LSTM, multiple conv-LSTM layers can be stacked to build up a deep conv-LSTM neural network.

However, the spatio-temporal variables, such as demand intensity and travel time rate during one time interval, are 2D matrices (see Definition 1 and 2), thus a transformation function $\mathscr{F}^T$: $R^{M\times N} \to R^{M\times N\times 1}$ is employed to transfer the initial input matrices into 3D tensors by simply adding one dimension (see Fig. 3).

4.2. *Fusion convolutional LSTM (FCL-Net)*

In this section, we propose a novel *fusion convolutional LSTM network* (FCL-Net), which integrates spatio-temporary variables and non-spatial time-series variables into one DL architecture for short-term passenger demand forecasting under the on-demand ride service platform. The structure of the proposed FCL-Net is illustrated in Fig. 4. Conv-LSTM layers and convolutional operators are employed to capture characteristics of spatio-temporary variables, while LSTM layers are implemented for non-spatial time-series variables. To fuse these two categories of variables, techniques including repeating and transformation functions, are utilized in the structure. The repeating function is denoted as $\mathscr{F}^R(\cdot;M,N)$: $R \to R^{M\times N\times 1}$, where $(\mathscr{F}^R(x;M,N))_{m,n,1} = x$ for any $m \in (1,2,...,M),n \in (1,2,...,N)$. It is also worth mentioning that the transformation function $\mathscr{F}^T$ should be applied to transfer 2D matrices $\boldsymbol{D}_t,\boldsymbol{\Gamma}_t$ to 3D tensors $\mathscr{D}_t,\Gamma_t \in R^{I\times J\times 1}$ to meet the consistent requirement of convolutional operators.

As mentioned in Problem 1, the forecasting target is the demand intensity during the tth time interval, which is denoted as $\mathscr{X}_t = \mathscr{D}_t$.

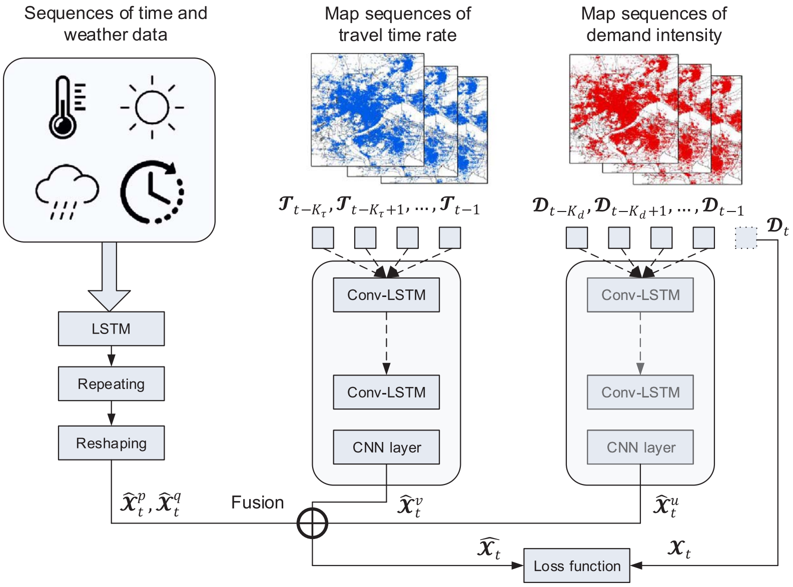

Sequences of time and weather data
Map sequences of travel time rate
Map sequences of demand intensity
$\mathcal{T}_{t-K_\tau},\mathcal{T}_{t-K_\tau+1},\dots,\mathcal{T}_{t-1}$
$\mathcal{D}_{t-K_d},\mathcal{D}_{t-K_d+1},\dots,\mathcal{D}_{t-1}$
$\mathcal{D}_t$
LSTM
Repeating
Reshaping
Conv-LSTM
Conv-LSTM
CNN layer
Conv-LSTM
Conv-LSTM
CNN layer
$\widehat{\mathcal{X}}_t^p,\widehat{\mathcal{X}}_t^q$
Fusion
$\widehat{\mathcal{X}}_t^v$
$\widehat{\mathcal{X}}_t^u$
$\widehat{\mathcal{X}}_t$
$\mathcal{X}_t$
Loss function

Fig. 4. Framework of the proposed FCL-Net approach.

4.2.1. Structure for spatio-temporary variables

Among the variables utilized in this paper, the historical demand intensity and travel time rate are spatio-temporary variables, as denoted in Definition 2. By considering that the historical demand intensity and travel time rate influence the future demand intensity in different ways, these two kinds of variables are fed into two separate architectures, each of which consists of a series of stacked conv-LSTM layers and convolutional operators. Suppose K_d,K_τ are the look-back time windows, L_d,L_τ are the number of stacked conv-LSTM layers, of the demand intensity and travel time rate, respectively, the formulations of the architecture for spatio-temporal variables are given as Eq. (13)–(16). The look-back time window refers to the number of previous hours taken as features for each sample, and equals to the input time step for each sample in a LSTM-based structure.

$$\left(\mathscr{U}_{t-K_d}^{(L_d)},\mathscr{U}_{t-K_d+1}^{(L_d)},\ldots,\mathscr{U}_{t-1}^{(L_d)}\right) = \mathscr{F}_{L_d}^{CL}\cdots\mathscr{F}_{l}^{CL}\cdots\mathscr{F}_{1}^{CL}\left(\mathscr{D}_{t-K_d},\mathscr{D}_{t-K_d+1},\ldots,\mathscr{D}_{t-1}\right) \tag{13}$$

$$\widehat{\mathscr{X}}_t^{u} = \sigma(W_{ux}*\mathscr{U}_{t-1}^{(L_d)} + b_u) \tag{14}$$

$$\left(\mathscr{V}_{t-K_\tau}^{(L_\tau)},\mathscr{V}_{t-K_\tau+1}^{(L_\tau)},\ldots,\mathscr{V}_{t-1}^{(L_\tau)}\right) = \mathscr{F}_{L_\tau}^{CL}\cdots\mathscr{F}_{l}^{CL}\cdots\mathscr{F}_{1}^{CL}\left(\mathit{\Gamma}_{t-K_\tau},\mathit{\Gamma}_{t-K_\tau+1},\ldots,\mathit{\Gamma}_{t-1}\right) \tag{15}$$

$$\widehat{\mathscr{X}}_t^{v} = \sigma(W_{vx}*\mathscr{V}_{t-1}^{(L_\tau)} + b_v) \tag{16}$$

where $\mathscr{U}_{t-k}^{(L_d)},k = 1,2,\ldots,K_d,\mathscr{V}_{t-\tau}^{(L_d)},k = 1,2,\ldots,K_\tau$ are the output hidden tensors in the highest-level layers of the architectures of demand and travel time rate, respectively. W_{ux},W_{vx} are convolutional operators utilized to further capture the spatial correlations of the highest-level output tensors, while b_u,b_v are the intercept parameters. Through these two structures, two high-level components $\widehat{\mathscr{X}}_t^{u},\widehat{\mathscr{X}}_t^{v}$ can be obtained, which will be further substituted into the fusion layer.

4.2.2. Structure for non-spatial time-series variables

Time variables (including the time-of-day and day-of-week) and weather variables (temperature, humidity, weather state, wind speed, and visibility) are the two classes of non-spatial time-series variables. Considering that time variables and weather variables affect the future demand intensity in different ways, we define two sequences of vectors: $e_s = (h_s,w_s),a_s = (at_s,ah_s,as_s,aw_s,av_s),s = 1,2,\ldots,t$. These two sequences of vectors are fed into two separate stacked LSTM architectures, which produce the two high-level components $\widehat{\mathscr{X}}_t^{p}$ and $\widehat{\mathscr{X}}_t^{q}$.

$$\left(p_{t-K_e+1}^{(L_e)},\ldots,p_{t}^{(L_e)}\right) = \mathscr{F}_{L_e}^{L}\cdots\mathscr{F}_{l}^{L}\cdots\mathscr{F}_{1}^{L}\left(e_{t-K_e+1},\ldots,e_{t-1},e_t\right) \tag{17}$$

$$\widehat{\mathscr{X}}_t^{p} = \mathscr{F}^{T}(\mathscr{F}^{R}(\sigma(w_p p_t^{(L_e)} + b_p))) \tag{18}$$

$$\left(q_{t-K_a+1}^{(L_a)},\ldots,q_{t-1}^{(L_a)}\right) = \mathscr{F}_{L_a}^{L}\cdots\mathscr{F}_{l}^{L}\cdots\mathscr{F}_{1}^{L}\left(a_{t-K_a},\ldots,a_{t-1}\right) \tag{19}$$

$$\widehat{\mathscr{X}}_t^{q} = \mathscr{F}^{T}(\mathscr{F}^{R}(\sigma(w_q q_{t-1}^{(L_a)} + b_q))) \tag{20}$$

where $p_{t-k}^{(L_e)},k = 1,2,\ldots,K_e,q_{t-K}^{(L_a)},k = 1,2,\ldots,K_a$ are the output hidden vectors in the highest LSTM layers L_e,L_a. K_e,K_a are the look-back time window of time features and weather features (non-spatial time-series features), while L_e,L_a are the number of designed LSTM layers for time and weather features respectively. w_p,w_q,b_p,b_q are the weighted and intercept parameters.

4.2.3. Fusion

Inspired by the fact that the high-level components have different contributions to the prediction, we employ Hadamard product '∘' to multiply these components by the four parameter matrices W_u,W_v,W_p and W_q, which can be learnt to evaluate the importance of the components during the training process. Therefore, the estimated demand intensity during the *t*th time interval is given by

$$\widehat{\mathscr{X}_t} = W_u\circ\widehat{\mathscr{X}}_t^{u} + W_v\circ\widehat{\mathscr{X}}_t^{v} + W_p\circ\widehat{\mathscr{X}}_t^{p} + W_q\circ\widehat{\mathscr{X}}_t^{q} \tag{21}$$

4.2.4. Objective function

During the training process of the FCL-Net, the object is to minimize the mean squared error between the estimated and real demand intensity, through which the weighted and intercept parameters can be learnt. The objective function of the architecture is shown in Eq. 22.

$$\min_{w,b}\ \|\mathscr{X}_t-\widehat{\mathscr{X}_t}\|_2^2 + \alpha\ \|W\|_2^2 \tag{22}$$

The second term of the objective function represents an L2-norm regularization term, which helps avoid overfitting issues. W stands for all the weighted parameters in $\widehat{\mathscr{X}_t}$, and α refers to a regularization parameter which balances the bias-variance tradeoff.

The training steps of the FCL-Net is illustrated in Algorithm 1.

Algorithm 1. FCL-Net Training

```
Input   Observations of demand intensity $\{D_1,\ldots,D_n\}$ in training set
        Observations of demand intensity $\{\mathbf{\Gamma}_1,\ldots,\mathbf{\Gamma}_n\}$ in training set
```

```
    Observations of time-of-day $\{h_1,...,h_n\}$, day-of-week $\{w_1,...,w_n\}$ in training set
    Observations of weather variables $\{at_1,...,at_n\},\{ah_1,...,ah_n\},\{as_1,...,as_n\},\{aw_1,...,aw_n\},\{av_1,...,av_n\}$
    lookback-windows: $K_d,K_\tau,K_e,K_a$
Output FCL-Net with learnt parameters
procedure FCL-Net Training
    Initialize a null set: $L \leftarrow \varnothing$
    for all available time intervals $t\ (1 \leqslant t \leqslant n)$ do
        $\mathscr{S}_t^d \leftarrow [\mathscr{D}_{t-K_d},\mathscr{D}_{t-K_d+1},...,\mathscr{D}_{t-1}]$
        $\mathscr{S}_t^\tau \leftarrow [\Gamma_{t-K_\tau},\Gamma_{t-K_\tau+1},...,\Gamma_{t-1}]$
        $\mathscr{S}_t^e \leftarrow [e_{t-K_e},...,e_{t-1},e_t]$, where $e_s = (h_s,w_s)$
        $\mathscr{S}_t^a \leftarrow [a_{t-K_a},...,a_{t-1},a_t]$, where $a_s = (at_s,ah_s,as_s,aw_s,av_s)$ ▷where $\mathscr{S}_t^d,\mathscr{S}_t^\tau,\mathscr{S}_t^e,\mathscr{S}_t^a$ are the sets of different categories of
      explanatory variables in one observation.
        A training observation $(\{\mathscr{S}_t^d,\mathscr{S}_t^\tau,\mathscr{S}_t^e,\mathscr{S}_t^a\},\mathscr{D}_t)$ is put into $L$
    end for
    Initialize all the weighted and intercept parameters
    repeat
        Randomly extract a batch of samples $L^b$ from $L$
        Estimate the parameters by the minimizing the objective function shown in Eq. (22) within $L^b$
    until convergence criterion met
end procedure
```

4.3. Spatial aggregated random forest for feature selection

Random forest, first introduced by Breiman (2001), is one of the most powerful ensemble learning algorithms for regression problems. Consider a training set with m observations, $L = \{(X^{(1)},y^{(1)}),...,(X^{(m)},y^{(m)})\}$, where $X^{(i)} \in R^p$ is the ith observation of features, and $y^{(i)} \in R$ is the ith observation of label. The kth decision tree can be represented as f_k: $R^p \rightarrow R$. Firstly, K sub-sample sets, $L_1,...,L_K$, are randomly generated from the whole training set L using the bootstrap method (the same sample can be selected by several sub-sample sets) and used for training models. Secondly, the kth out-of-bag sample set, O_k, defined as the difference of set L and set L_k, is used for testing. Finally, the out-of-bag error of the kth tree, $errO_k$, is denoted as the average testing error in the set O_k (shown in Eq. (23)).

$$errO_k = \frac{1}{n}\sum_{i\in O_k} (y^{(i)}-\hat{y}^{(i)})^2 \tag{23}$$

where $\hat{y}^{(i)} = f_k(X^{(i)})$ is the estimated value of the ith label based on tree k. The out-of-bag error can be utilized to calculate the feature importance through the following steps (Genuer et al., 2015): (1) permute the jth variable of X in each O_k to get a new out-of-bag samples O_k'; (2) calculate the out-of-bag error, $\widetilde{errO}_k^j$ in the new sets of samples O_k'; (3) the importance of the j^{th} variable, $VI(X_j)$, is equal to the average difference between $errO_k$ and $\widetilde{errO}_k^j$ of all trees (shown in Eq. (24)).

$$VI(X_j) = \frac{1}{K}\sum_k (\widetilde{errO}_k^j - errO_k) \tag{24}$$

Considering that the dependent variables in the passenger demand forecasting, $\boldsymbol{D}_t \in R^{I\times J}$, form an $I\times J$ matrix, instead of a continuous value in the standard random forest, we develop a spatial aggregated random forest which consists of $I\times J$ standard random forests, to examine the aggregated variable importance partitioned by category and look-back time window. To illustrate the spatial aggregated random forest, we extend Problem 1 to Problem 2, given by

Problem 2. Given the historical observations and known information $\{d_s^{i,j},\tau_s^{i,j}|s = t-K,...,t-1, i \in \{1,...,I\}, j \in \{1,...,J\}; h_s,w_s|s = t-K+1,...,t; at_s,ah_s,as_s,aw_s,av_s|s = t-K,...,t-1\}$, predict $d_s^{i',j'}$ via the standard random forest $f^{i',j'}$, for all $i' \in \{1,...,I\}, j' \in \{1,...,J\}$. The length of the look-back window is denoted as K.

$VI^{i',j'}$ is denoted as the function to calculate variable importance in random forest $f^{i',j'}$. Two tensors, $\mathbb{V}^d,\mathbb{V}^\tau \in R^{I\times J\times I\times J\times K}$, where $(\mathbb{V}^d)_{i,j,i',j',k} = VI^{i',j'}(d_{t-k}^{i,j}),(\mathbb{V}^\tau)_{i,j,i',j',k} = VI^{i',j'}(\tau_{t-k}^{i,j})$, are denoted to store the variable importance of the two categories of spatial-temporal variables. $(\mathbb{V}^d)_{i,j,i',j',k}$ refers to the variable importance of the passenger demand in $\{i,j\}$ during time interval $t-k$ in the problem of forecasting passenger demand of $\{i',j'\}$ during time slot t. As for non-spatial time-series variables, we define $\mathbb{V}^h,\mathbb{V}^w,\mathbb{V}^{at},\mathbb{V}^{ah},\mathbb{V}^{as},\mathbb{V}^{aw},\mathbb{V}^{av} \in R^{I\times J\times K}$, where $(\mathbb{V}^h)_{i',j',k} = VI^{i',j'}(h_{t-k})$, and the same expression for w,at,ah,as,aw,av. All the variable importance in $\mathbb{V}^d,\mathbb{V}^\tau,\mathbb{V}^h,\mathbb{V}^w,\mathbb{V}^{at},\mathbb{V}^{ah},\mathbb{V}^{as},\mathbb{V}^{aw},\mathbb{V}^{av}$ is normalized to percentage via dividing each variable importance by the sum of all variable importance.

Firstly, we examine the variable importance partitioned by category, i.e. $\sum_i \sum_j \sum_{i'} \sum_{j'} \sum_k \mathbb{V}^d, \sum_{i'} \sum_{j'} \sum_k \mathbb{V}^h$, etc., to select the important variables in terms of category. Secondly, we investigate the variable importance partitioned by category and look-back time window $k\,(k \in \{1,2,...,K\})$, to select a suitable look-back window for each variable category.

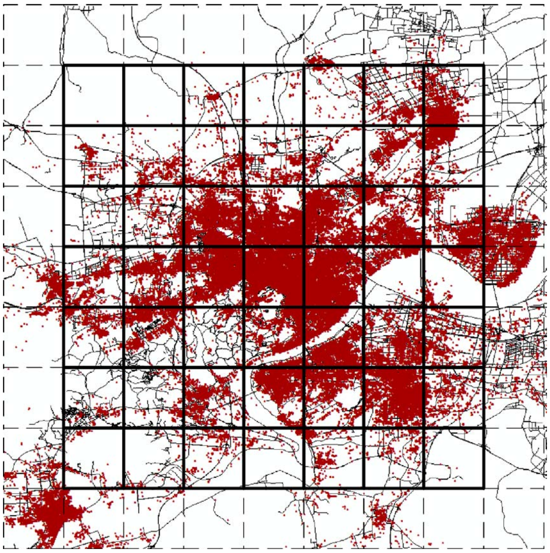


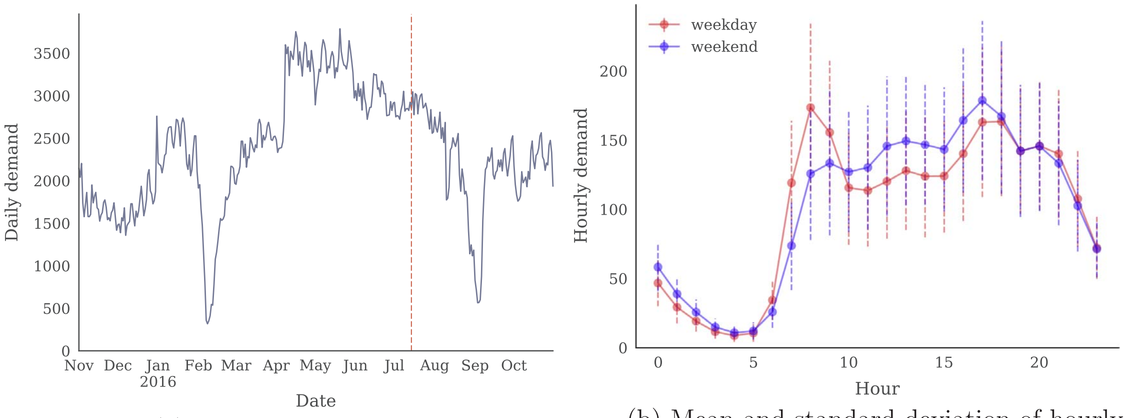
Fig. 5. The investigated region partitioned into 7 × 7 grids.

5. Experiments and results

5.1. On-demand ride service platform data

The datasets utilized in this paper are extracted from DiDi Chuxing, the largest on-demand ride service platform in China, during one-year period between November 1, 2015 and November 1, 2016. We randomly obtain 1,000,000 requesting orders from the platform, each of which consists of the requesting time, travel distance, travel time, longitude and latitude. Although these 1,000,000 samples are randomly selected with a sampled ratio of 10% due to the limitation of data availability, the random sampling could largely ensure the transfer from this dataset to the field application. In this dataset, the demand intensity can be calculated by summing up the orders starting from one region in one time interval; the average travel time rate can also be obtained from the record of travel distance and travel time with the definition in Section 3. The studied site is located in the downtown of Hangzhou, China, starting from 120.00° E to 120.35° E in longitude, and from 30.15° N to 30.45° N in latitude. The dataset is partitioned into 1-h time intervals, and the investigated region is partitioned into 7 × 7 grids, as shown in Fig. 5. Each grid is roughly a rectangle with length equaled to 4.77 km and width equaled to 4.81 km. We also collect one-hour aggregated weather variables, including temperature, humidity, weather state, wind speed, and visibility, during the same period from the China Meteorological Administration.

To avoid using future information, the dataset is divided into 70% training set comprised of observations between November 1, 2015 and July 14, 2016, and the 30% test set consisting of the remaining observations between July 15, 2016 and November 1, 2016. Fig. 6(a) shows the total demand of all girds in different days within the investigated training (before the red dash line) and testing (after the red dash line) period. It can be observed that both the training and test sets have some periods when the total demand is



(a) Demand in different days

(b) Mean and standard deviation of hourly demand

Fig. 6. Temporal variations of the on-demand ride services.

abnormally low. The first low-demand period is due to the Chinese Lunar New Year in the beginning of February (in the training set), while the second one is affected by the G20 summit held in the beginning of September (in the test set). These abnormal periods raise challenges for the short-term demand forecasting.

It can be observed from Fig. 6(b), which shows the mean and variance of passenger demand in different hours of a day based on the training set, that the passenger demand in weekdays demonstrates a double-peak nature while the passenger demand in weekends shows a single-peak property. Therefore, the peak hours, off-peak hours, and sleep hours are separately defined for weekdays and weekends.

5.1.1. Exploring the spatio-temporal correlations

The reason for utilizing a tailored spatio-temporal DL architecture is that there exist spatio-temporal correlations among the spatio-temporal variables, i.e., the demand intensity and travel time rate. To validate this assumption, we examine the correlations between the demand intensity at the tth time interval and spatio-temporal variables ahead of the tth time interval by employing the Pearson correlation, given by

$$Corr(Y,Z) = \frac{E[(Y-E(Y))'(Z-E(Z))]}{E[(Y-E(Y))^2]E[(Z-E(Z))^2]} \tag{25}$$

where Y,Z are two random variables with the same number of observations.

Firstly, we calculate the Pearson correlations between the demand intensity at time t in grid (i',j') and demand intensity, travel time rate at $t-k$ time interval in grid (i,j), for all $i,i' \in \{1,...,I\}, j,j' \in \{1,...,J\}, k \in \{1,2,3,4\}$. Secondly, we average these correlations partitioned by spatial distances and look-back time intervals. The spatial distance of grid (i,j) and (i',j') is denoted as the Euclidean distance between the central points of the two grids.

Fig. 7 shows the average correlations between the dependent variables (the demand intensity at time t in grid (i',j')) and the explanatory variables (the demand intensity and travel time rate at time $t-k$ in grid (i,j)). It can be observed that the average correlations drop gradually but not sharply, with the increase of the spatial distance, indicating that there exit strong spatial correlations between each grid and its neighbors. On the other hand, it is not surprising that variables with shorter look-back time intervals have higher correlations, but the variables with large look-back time intervals are also correlated with the to-be-predicted demand intensity to some extent. This correlation analysis of the dataset provides evidence that the spatial and temporal dependencies exist among the spatio-temporal variables.

5.2. Feature selection

Firstly, Fig. 8 shows the variable importance partitioned by category, based on our proposed spatial aggregated random forest algorithm. It can be observed that the two categories of spatial-temporary variables, travel time rate, and demand intensity, are the dominating factors, followed by time-of-day and temperature. However, other variables, such as day-of-week, humidity, etc., have few contributions (less than 5%) to the prediction.

Secondly, Table 1 presents the relative importance of the variables sorted by category and look-back time interval ("-" represents not applicable). Fig. 9 displays the top 20 important variables. We set the look-back time window K to be 8 for all category of variables. It can be found that the time-of-day during time interval t is the most important variable, followed by the demand intensity and travel time rate during $t-1$ time interval. The importance of all kinds of variables decreases with the look-back time window, but different categories of variables show different descent speeds. The travel time rate far before t still has considerable variable importance, while the time-of-day prior to time t makes little contribution.

Selecting appropriate categories of variables and the suitable look-back window helps to improve computation efficiency with

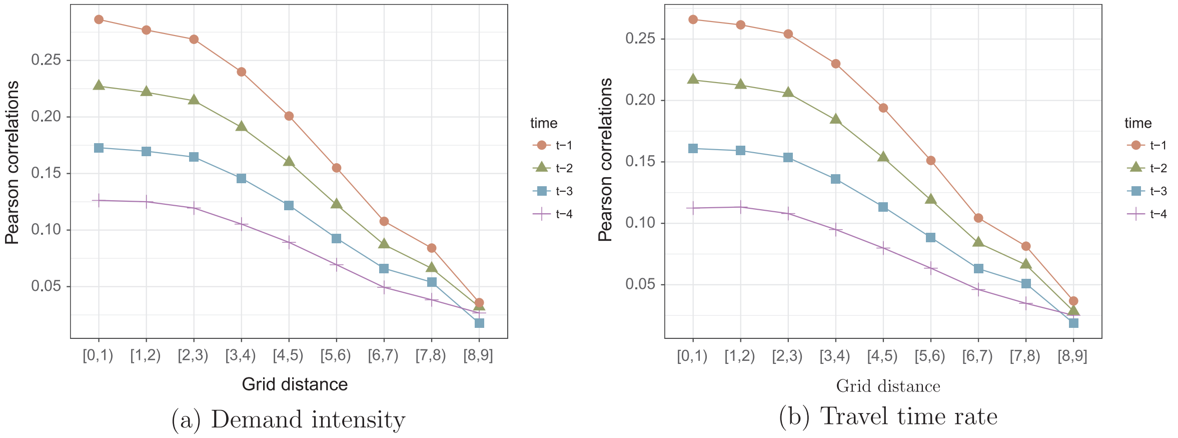


Fig. 7. Average correlations partitioned by distance and time.

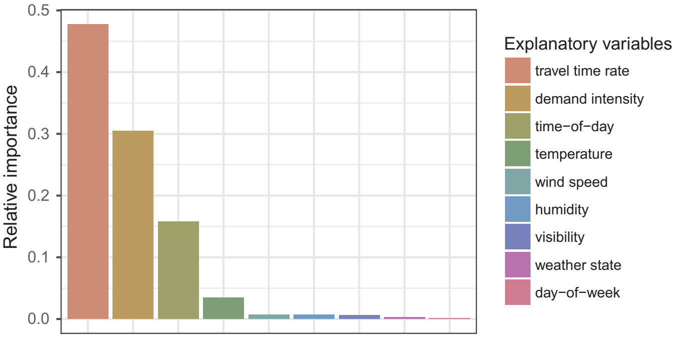


Fig. 8. Variable importance ranking by the random forest.

Table 1
The relative importance of variables partitioned by category and look-back time interval (%).

s	D_s	Γ_s	at_s	ah_s	as_s	aw_s	av_s	h_s	w_s
$t-8$	2.628	5.005	0.589	0.074	0.036	0.080	0.089	–	–
$t-7$	2.040	4.432	0.275	0.084	0.035	0.086	0.076	0.137	0.041
$t-6$	1.978	4.790	0.251	0.088	0.032	0.083	0.072	0.039	0.018
$t-5$	2.280	4.852	0.223	0.071	0.032	0.074	0.091	0.013	0.013
$t-4$	2.895	4.984	0.205	0.085	0.043	0.079	0.089	0.021	0.022
$t-3$	3.140	6.038	0.333	0.097	0.033	0.069	0.071	0.514	0.013
$t-2$	3.518	7.008	0.450	0.110	0.039	0.172	0.070	0.589	0.022
$t-1$	11.999	10.658	1.105	0.116	0.039	0.083	0.076	0.671	0.015
t	–	–	–	–	–	–	–	13.808	0.012

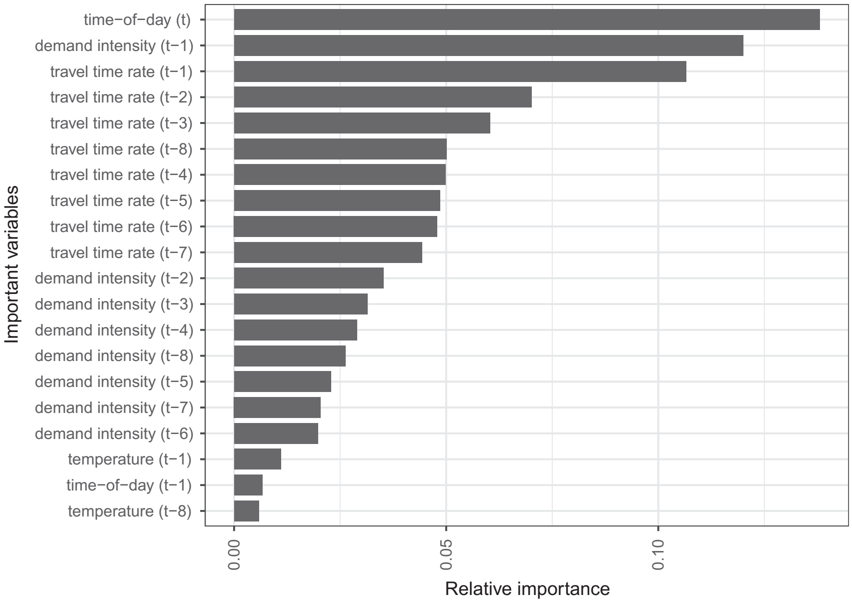


Fig. 9. Top 20 important variables partitioned by category and time.

little loss of predictive performance. In this paper, by considering the trade-off between computation efficiency and predictive performance, we select 4 categories of variables: demand intensity, travel time rate, time-of-day, and temperature, with 4,8,2,2 look-back time windows, respectively. This feature selection reduces the number of variables in each observation from $8 \times 7 \times 7 \times 2 + 8 \times 5 + 8 \times 2 = 840$ to $8 \times 7 \times 7 + 4 \times 7 \times 7 + 2 + 2 = 592$.

5.3. Model comparisons

The proposed FCL-Net with full variables and selected variables (selected by the spatial aggregated random forest) are trained on the training set and validated on the test set, respectively. Meanwhile, the conv-LSTM network, which is only fed with the historical demand intensity, is trained and tested in the same way. The definition of the aforementioned three models are shown as follows:

(1) *Conv-LSTM with only historical demand intensity*: This model utilizes historical observations of demand intensity $\{D_s|s = t-K_d,...,t-1\}$ to predict future demand intensity $\boldsymbol{D}_t$, where $K_d = 8$ in this paper. The architecture of Conv-LSTM is introduced in Section 4.2.
(2) *FCL-Net with full variables*: Historical observations of all variables $\{D_s,\boldsymbol{\Gamma}_s|s = t-K,...,t-1;h_s,w_s|s = t-K+1,...,t;at_s,ah_s,as_s,aw_s,av_s|s = t-K,...,t-1\}$, where $K = 8$, are utilized to predict future demand intensity $\boldsymbol{D}_t$. The training process of this model is illustrated in Algorithm 1.
(3) *FCL-Net with selected variables*: This model utilizes the historical observations of the selected variables $\{D_s|s = t-K_d,...,t-1,\boldsymbol{\Gamma}_s|s = t-K_\tau,...,t-1;h_s|s = t-K_h+1,...,t;at_s|s = t-K_{at},...,t-1\}$ where $K_d = 4,K_\tau = 8,K_h = 2,K_{at} = 2$, to forecast future demand intensity $\boldsymbol{D}_t$. The training process of this model is the same as Algorithm 1 except that the inputs are replaced by the selected variables.

In this section, the structure of FCL-Net is comprised of 4 Conv-LSTM layers for spatio-temporal features and 4 LSTM layers for non-spatial time-series features. Each Conv-LSTM cell consists of 40 filters/channels to fully catch up spatial information. The number of training epochs is set to be 100, while the batch size is set to be 10.

Apart from the proposed three models, several benchmark algorithms are also tested. The benchmark algorithms include three traditional time-series forecasting models (i.e., HA, MA, and ARIMA) and several state-of-the-art machine learning/deep learning approaches (i.e., XGBoost, ANN, CNN, and LSTM).

(1) *HA*: The historical average model predicts the future demand intensity in the test set based on the empirical statistics in the training set. For example, the average demand intensity during 8–9 AM in grid (i,j) is estimated by the mean of all historical demand intensity during 8–9 AM in grid (i,j).
(2) *MA*: The moving average model is widely-used in time-series analysis, which predicts future value by the mean of serval nearest historical values. In this paper, the average of 8 previous demand intensities in grid (i,j) are used to predict the future demand intensity in grid (i,j).
(3) *ARIMA*: The autoregressive integrated moving average model (Box and Pierce, 1970) integrates the autoregressive (AR), integrated (I), and MA parts, and considers trends, cycles, and non-stationary characteristics of a dataset simultaneously.
(4) *ANN*: The artificial neural network (Rumelhart et al., 1985, 1988) employs all the variables, including the historical demand intensity, travel time rate, hour and week state, and weather variables, with look-back time window $K = 8$, of a specific grid (i,j), to predict the future demand intensity in grid (i,j). ANN does not differentiate variables across time and thus fails to capture time dependencies.
(5) *LSTM*: In LSTM (Hochreiter and Schmidhuber, 1997), all the variables in grid (i,j) are reshaped to a matrix with one axis as time step (the size of which equals look-back time window $K = 8$) and another axis as the feature category. In this way, all the features utilized in FCL-Net are fed into LSTM for training. LSTM considers temporal dependencies, but does not capture spatial dependencies.
(6) *CNN*: In the convolutional neural network(Krizhevsky et al., 2012), the demand intensity and travel time rate of all grids in each time step (e.g., $t-1$ time step) is viewed as a picture, thus $8 \times 2 = 16$ pictures are obtained for each sample. We build 16 convolutional neural networks for spatio-temporal features and a fully-connected neural network for the non-spatial time-series features, which are then fused together in the top layer.
(7) *XGBoost*: XGBoost(Chen and Guestrin, 2016) is a scalable end-to-end tree boosting system, which is proven to achieve outstanding performance in a board range of machine learning challenges. All the variables utilized in FCL-Net (including spatio-temporal features and non-spatial time-series features) are reshaped to a vector and fed into XGBoost for training.

To ensure fairness, the aforementioned benchmark machine learning algorithms have the same input features (same category and look-back time windows) as the FCL-Net, while the three traditional time-series models make use of the whole time-series of the historical demand intensities. The deep learning approaches (ANN, LSTM, and CNN) are trained with 100 epochs, which is consistent with FCL-Net. All the benchmarks are under fine-tuned. Before model training and validation, the demand intensity, travel time rate, hour-of-day, day-of-week, and weather variables, are standardized to the range [0,1], through the max-min standardization, respectively.

Our experiment platform is a server with 20 CPU cores (Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20 GHz), 251 GB RAM, and one GPU (NVIDIA UNIX x86-64 Kernel Module 375.66). We use python 2.7.5 with scikit-learn (Pedregosa et al., 2011), tensorflow (Abadi et al., 2015), keras (Chollet, 2015), and XGBoost (Chen and Guestrin, 2016) on CentOS Linux release 7.2.1511 (Core) for comparing the models.

We evaluate the models via the four measures of effectiveness: root mean squared error (RMSE), coefficient of determination (R^2), mean absolute error (MAE), and mean absolute percentage error (MAPE), given by

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y^{(i)}-\hat{y}^{(i)})^2} \tag{26}$$

Table 2
Predictive performance comparison.

Model	RMSE	R^2	MAE	MAPE@10[a]	Time (min)
HA	0.0378	0.736	0.0192	28.06%	0.01
MA	0.0511	0.518	0.0260	43.63%	0.01
ARIMA	0.0345	0.780	0.0178	30.40%	2.27
XGBoost	0.0322	0.801	0.0176	27.34%	2.13
ANN	0.0331	0.797	0.0198	27.95%	17.38
LSTM	0.0332	0.798	0.0172	31.27%	193.39
CNN	0.0175	0.773	0.0106	27.52%	85.42
Conv-LSTM (demand) [b]	0.0315	0.806	0.0175	26.18%	99.93
FCL-Net (selected) [c]	0.0163	0.803	0.0096	20.46%	169.90
FCL-Net (full) [d]	0.0160	0.812	0.0094	21.79%	224.71

[a] MAPE in the testing samples with demand intensity greater or equal to 10.
[b] Conv-LSTM with only demand intensity as input features.
[c] FCL-Net with selected variables as input.
[d] FCL-Net with full variables as input.

$$\mathrm{R}^2 = 1-\frac{\sum_{i=1}^{n}(y^{(i)}-\hat{y}^{(i)})^2}{\sum_{i=1}^{n}(y^{(i)}-\bar{y})^2} \tag{27}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y^{(i)}-\hat{y}^{(i)}| \tag{28}$$

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y^{(i)}-\hat{y}^{(i)}|}{y^{(i)}} \tag{29}$$

where $y^{(i)},\hat{y}^{(i)}$ are the ith ground truth and estimated value of the demand intensity, respectively. $\bar{y}$ is the mean of all $y^{(i)}$, and n is the size of the test set. We use RMSE, (R^2), and MAE to measure the total predictive accuracy/degree of fitting in the whole test data, and use MAPE@10 (MAPE in the samples with demand intensity greater or equal to 10) to measure models' predictive performance in the highly demanded regions and time periods. The platform would pay more attention to the predicted highly demanded regions in peak hours, with which it can dispatch vacant drivers to those regions. In this paper, MAPE@10 covers the top 4.45% largest samples in the test set.

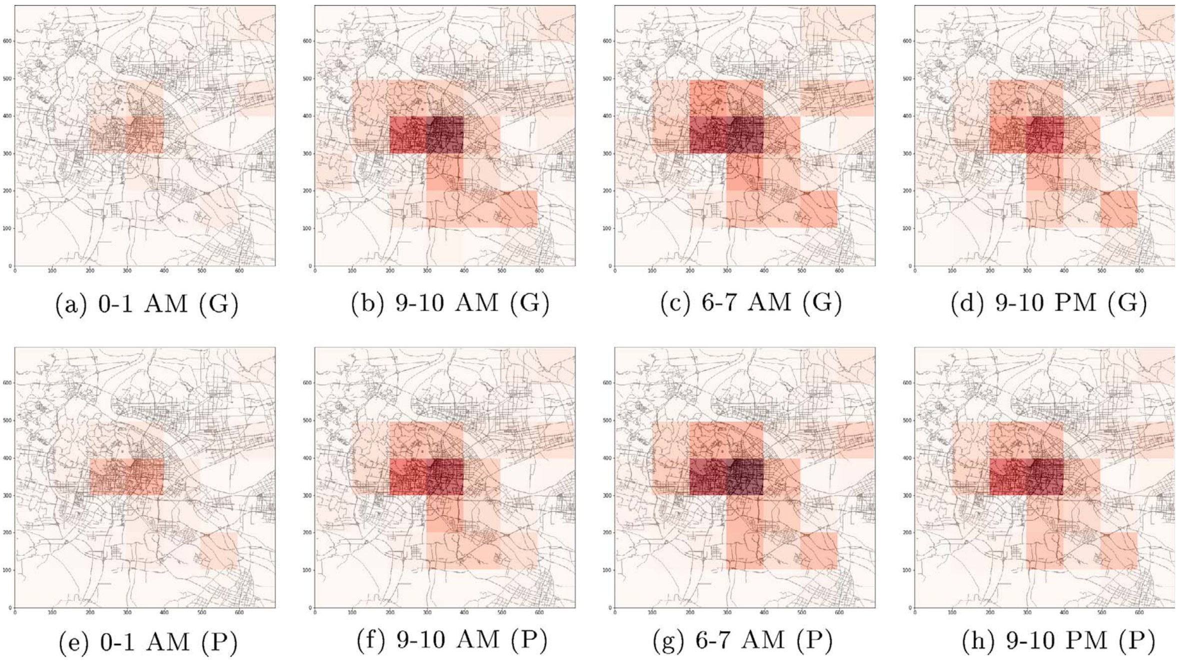
Table 2 shows the predictive performance of the proposed models and benchmark models on the test set. It can be found that the proposed FCL-Nets outperform the benchmarks in the four measurements of predictive performance. The FCL-Net with full variables achieves the best predictive performance measured by RMSE (0.016), which is 8.6% lower than the best benchmark CNN (0.0175). With the feature selection, FCL-Net achieves a 24.4% drop on the training time, while bearing only a 1.8% decrease on predictive performance measured by RMSE. The FCL-Net with selected variables (with MAPE@10 equaled to 20.46%) even outperforms the FCL-Net with full variables (with MAPE@10 equaled to 20.46%) in the samples with a large demand intensity. It indicates that feature selection is valuable to FCL-Net since it takes into account the trade-off between the computation complexity and predictive performance.

It is also interesting to find that both FCL-Nets have relatively 48.3% lower RMSE than Conv-LSTM with only the historical demand intensity, which indicates that the exogenous variables make great contribution to the short-term passenger demand forecasting. From the comparisons among the benchmark models, we can see that the machine learning approaches have better predictive performance but longer computing time than the traditional time-series models. CNN which captures spatial correlations outperforms LSTM which only models the temporal characteristics, providing another verification that the consideration of spatial correlations is of great importance to the city-wide demand forecasting.

Fig. 10 shows some samples of heat maps of the ground truth passenger demand and predicted results by FCL-Net, where the deeper color implies a larger demand intensity. It is obvious that the demand intensity in peak hours (e.g., 9–10 AM and 6–7 PM) is much higher than that in sleep hours (e.g., 0–1 AM). The demand intensity is unbalanced across space: the central grids have much a higher demand intensity than other grids. The trend of the demand intensity over time is even different in different grids and different days, which makes it hard to forecast short-term passenger demand. From the samples of visualization, we can find that the FCL-Net primarily captures the spatio-temporal characteristics of the demand intensity and makes accurate forecasting. The combination of short-term passenger demand forecasting and visualization helps traffic operators of the platform/government to detect and forecast grids with oversupply and overfull demand and design proactive strategies to avoid these imbalanced conditions.

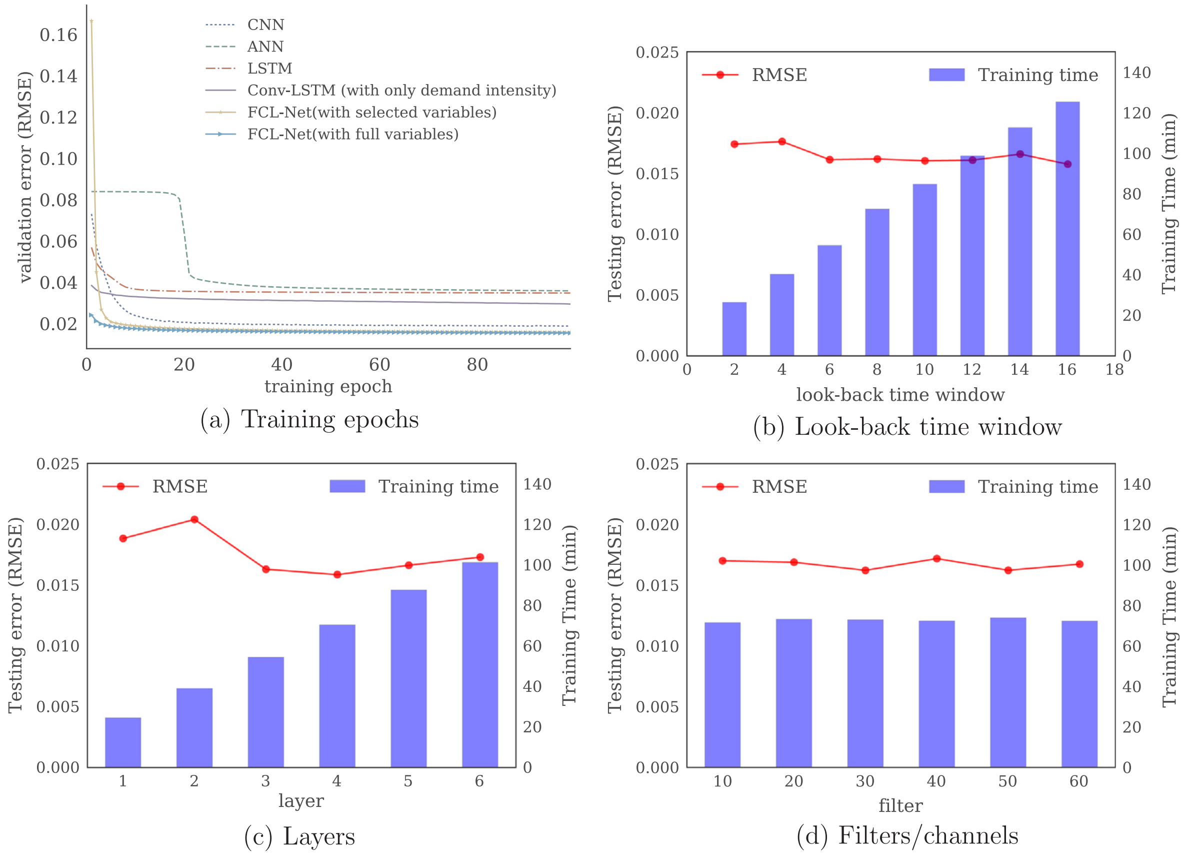
5.4. Sensitivity analysis

In this section, we conduct the sensitivity analysis and parameter tuning on FCL-Net, where four kinds of parameters are investigated, including the number of training epochs, look-back time windows, layers of the structure, and filters of convolutional units. Firstly, the validation error after each training epoch is recorded for FCL-Net and the benchmark models. From Fig. 11(a), it can


(a) 0-1 AM (G) (b) 9-10 AM (G) (c) 6-7 AM (G) (d) 9-10 PM (G)

(e) 0-1 AM (P) (f) 9-10 AM (P) (g) 6-7 AM (P) (h) 9-10 PM (P)

Fig. 10. Comparison of the ground truth (G) and predicted passenger demand by FCL-Net (P).



(a) Training epochs (b) Look-back time window

(c) Layers (d) Filters/channels

Fig. 11. Parameter tuning and sensitivity analysis of FCL-Net.

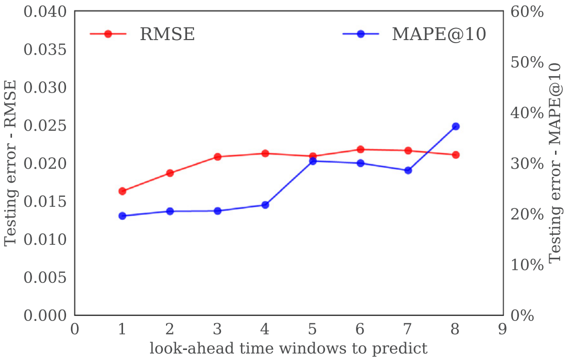


Fig. 12. Sensitivity analysis on multi-step prediction.

be observed that the RMSE of FCL-Nets drops quickly in the initial 5 epochs and then decreases gradually, which indicates that FCL-Nets have a high convergence speed. It is reported in Section 5.3 that the FCL-Net with full variables can cost 224.71 min for 100 epochs' training, which seems to be computationally expensive. However, the number of training epochs can be reduced to 10 or even 5, with the predictive performance slightly sacrificed, under the case where high-efficient computation resources are not reachable.

With the knowledge of the training convergence in mind, we then train the FCL-Net with full variables under different look-back time windows, layers, and filters/channels, with 30 training epochs (which are enough for convergence). Fig. 11(b) shows that the RMSE of FCL-Net gradually decreases while the training time linearly increases with the increase of the look-back time window. This result is intuitive: the more temporal information is utilized, the better predictive performance can be achieved but the longer training time is needed. Fig. 11(c) demonstrates that the RMSE first drops and then rises with the increase of Conv-LSTM layers in the structure of FCL-Net, while the computing time monotonously increases with the number of layers. With the increase of the depth of FCL-Net, it could face overfitting issues, which can explain the rise of its RMSE when the number of layers is greater than 4. From Fig. 11(d), it is interesting to find that both the training error (RMSE) and training time have no significant relationships with the number of filters/channels of convolutional units. This phenomenon could be explained as that the DL backend *tensorflow* is distributed, thus the extracted features in each filter/channel of FCL-Net's convolutional units can be calculated in parallel.

The sensitivity analysis on the multi-step prediction is also conducted in this section. The x-axis in Fig. 12 refers to the look-ahead time window (n) to predict, which indicates predicting the demand intensity of $t + n$ time interval with the information prior to time interval t, while the two y-axes record the RMSE and MAPE@10 in different n, respectively. The results show that the worst RMSE is 0.0218, which is lower than most of the baselines on predicting the $t + 1$ demand intensity. The model can also maintain a MAPE@10 lower than 30% within 7 look-ahead time windows. These results show that the predictive accuracy of FCL-Net deteriorates slowly over the look-ahead time window, which means the proposed method adapts well to the multi-step prediction tasks.

6. Conclusions

In this paper, we propose a DL approach, named the fusion convolutional LSTM (FCL-Net), for the short-term passenger demand forecasting under an on-demand ride service platform. The proposed architecture is fused by multiple conv-LSTM layers, LSTM layers, and convolutional operators, and fed with a variety of explanatory variables including the historical passenger demand, travel time rate, time-of-day, day-of-week, and weather conditions. A tailored spatially aggregated random forest is employed to rank the importance of the explanatory variables. The ranking is then used for feature selection. We trained two FCL-Nets, i.e., one trained with full variables, and the other trained with selected variables. In addition, the conv-LSTM which only takes historical passenger demand as the explanatory variable is also established. These three models are compared with several benchmark algorithms including the HA, MA, ARIMA, XGBoost, ANN, LSTM, and CNN. The models are validated on the real-world data provided by DiDi Chuxing, the results of which show that the two FCL-Nets outperform the benchmark algorithms in the measurements of RMSE, R-square, MAE, and MAPE in large samples, indicating that the proposed approach performs better at capturing the spatio-temporal characteristics for the short-term passenger demand forecasting. The FCL-Net utilizing full variables outperforms the best benchmark CNN by 8.6% in RMSE. The feature selection process helps FCL-Net reduce training time by 24.4% while only suffering from the 1.8% loss in the predictive performance (measured by RMSE). The experiments results also show that the consideration of exogenous variables is important since the FCL-Net achieves a 48.3% lower RMSE compared to Conv-LSTM with only the historical demand intensity. The evidence from benchmark models proves that characterizing spatial correlations in models can greatly improve the predictive accuracy.

This paper explores the short-term passenger demand forecasting under the on-demand ride service platform via a novel spatio-temporal DL approach. Accurate real-time passenger demand forecasting can provide suggestions for the platform to rebalance the spatial distribution of cruising cars to meet passenger demand in each region, which will improve the car utilization rate and passengers' degree of satisfaction. One limitation of this paper is that the dataset obtained is randomly sampled with the 10% sampling ratio, thus there might exist bias between the studied dataset and field application. The proposed model will be tested with more data in future research. In particular, we anticipate to evaluate the proposed methods in the field application on on-demand ride service platforms in the future. Furthermore, to understand the complex interactions among the variables in the on-demand ride service market is far beyond predicting passenger demand. In the following studies, we expect to utilize more economic analyses and machine learning techniques to further explore the relationship between the endogenous and exogenous variables under on-demand ride service platforms.

Acknowledgements

This research is financially supported by Zhejiang Provincial Natural Science Foundation of China [LR17E080002], Key Laboratory of Road & Traffic Engineering of the Ministry of Education [TJDDZHCX004], National Natural Science Foundation of China [51508505, 71771198, 51338008], and Fundamental Research Funds for the Central Universities [2017QNA4025]. The authors are grateful to DiDi Chuxing (htpp://www.xiaojukeji.com) for providing us some sample data.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from < http://tensorflow.org/ >.

Ahmed, M., Abdel-Aty, M., 2013. A data fusion framework for real-time risk assessment on freeways. Transp. Res. Part C: Emerg. Technol. 26, 203–213.

Alonso-Mora, J., Samaranayake, S., Wallar, A., Frazzoli, E., Rus, D., 2017. On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. Proc. Nat. Acad. Sci. 201611675.

Antoniou, C., Koutsopoulos, H.N., Yannis, G., 2013. Dynamic data-driven local traffic state estimation and prediction. Transp. Res. Part C: Emerg. Technol. 34, 89–107.

Bachmann, C., Abdulhai, B., Roorda, M.J., Moshiri, B., 2013. A comparative assessment of multi-sensor data fusion techniques for freeway traffic speed estimation using microsimulation modeling. Transp. Res. Part C: Emerg. Technol. 26, 33–48.

Boto-Giralda, D., Díaz-Pernas, F.J., González-Ortega, D., Díez-Higuera, J.F., Antón-Rodríguez, M., Martínez-Zarzuela, M., Torre-Díez, I., 2010. Wavelet-based denoising for traffic volume time series forecasting with self-organizing neural networks. Comput.-Aided Civil Infrastruct. Eng. 25, 530–545.

Box, G.E., Pierce, D.A., 1970. Distribution of residual autocorrelations in autoregressive-integrated moving average time series models. J. Am. Stat. Assoc. 65, 1509–1526.

Breiman, L., 2001. Random forests. Mach. Learn. 45, 5–32.

Chan, K.Y., Dillon, T.S., Singh, J., Chang, E., 2012. Neural-network-based models for short-term traffic flow forecasting using a hybrid exponential smoothing and levenberg–marquardt algorithm. IEEE Trans. Intell. Transp. Syst. 13, 644–654.

Chen, Q., Song, X., Yamada, H., Shibasaki, R., 2016. Learning deep representation from big and heterogeneous data for traffic accident inference. In: Thirtieth AAAI Conference on Artificial Intelligence.

Chen, T., Guestrin, C., 2016. Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, ACM. pp. 785–794.

Chen, X., Zahiri, M., Zhang, S., 2017. Understanding ridesplitting behavior of on-demand ride services: an ensemble learning approach. Transp. Res. Part C: Emerg. Technol. 76, 51–70.

Cheng, Q., Liu, Y., Wei, W., Liu, Z., 2016. Analysis and forecasting of the day-to-day travel demand variations for large-scale transportation networks: a deep learning approach. doi:http://dx.doi.org/10.13140/RG.2.2.12753.53604.

Chollet, F., et al., 2015. Keras. < https://github.com/fchollet/keras >.

Deng, Z., Ji, M., 2011. Spatiotemporal structure of taxi services in shanghai: Using exploratory spatial data analysis. In: The 19th IEEE International Conference on Geoinformatics, pp. 1–5.

Fei, X., Lu, C.C., Liu, K., 2011. A Bayesian dynamic linear model approach for real-time short-term freeway travel time prediction. Transp. Res. Part C: Emerg. Technol. 19, 1306–1318.

Genuer, R., Poggi, J.M., Tuleau-Malot, C., 2015. Vsurf: An R package for variable selection using random forests. The R Journal 7, 19–33.

Ghosh, B., Basu, B., O'Mahony, M., 2009. Multivariate short-term traffic flow forecasting using time-series analysis. IEEE Trans. Intell. Transp. Syst. 10, 246–254.

Guo, J., Huang, W., Williams, B.M., 2014. Adaptive kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification. Transp. Res. Part C: Emerg. Technol. 43, 50–64.

He, F., Shen, Z.J.M., 2015. Modeling taxi services with smartphone-based e-hailing applications. Transp. Res. Part C: Emerg. Technol. 58, 93–106.

Hochreiter, S., 1991. Untersuchungen zu dynamischen neuronalen Netzen (Ph.D. thesis). diploma thesis, institut für informatik, lehrstuhl prof. brauer, technische universität münchen.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9, 1735–1780.

Huang, S., Sadek, A.W., 2009. A novel forecasting approach inspired by human memory: the example of short-term traffic volume forecasting. Transp. Res. Part C: Emerg. Technol. 17, 510–525.

Huang, W., Song, G., Hong, H., Xie, K., 2014. Deep architecture for traffic flow prediction: deep belief networks with multitask learning. IEEE Trans. Intell. Transp. Syst. 15, 2191–2201.

Jiang, X., Zhang, L., Chen, X., 2014. Short-term forecasting of high-speed rail demand: a hybrid approach combining ensemble empirical mode decomposition and gray support vector machine with real-world applications in china. Transp. Res. Part C: Emerg. Technol. 44, 110–127.

Khashei, M., Bijari, M., Ardali, G.A.R., 2012. Hybridization of autoregressive integrated moving average (ARIMA) with probabilistic neural networks (PNNs). Comput. Ind. Eng. 63, 37–45.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105.

LeCun, Y., Haffner, P., Bottou, L., Bengio, Y., 1999. Object recognition with gradient-based learning. Shape Contour Group. Comput. Vision 1681, 823.

Lv, Y., Duan, Y., Kang, W., Li, Z., Wang, F.Y., 2015. Traffic flow prediction with big data: a deep learning approach. IEEE Trans. Intell. Transp. Syst. 16, 865–873.

Ma, X., Dai, Z., He, Z., Ma, J., Wang, Y., Wang, Y., 2017. Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction. Sensors 17, 818.

Ma, X., Tao, Z., Wang, Y., Yu, H., Wang, Y., 2015a. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. Transp. Res. Part C: Emerg. Technol. 54, 187–197.

Ma, X., Yu, H., Wang, Y., Wang, Y., 2015b. Large-scale transportation network congestion evolution prediction using deep learning theory. PloS one 10, e0119044.

Moreira-Matias, L., Gama, J., Ferreira, M., Damas, L., 2012. A predictive model for the passenger demand on a taxi network. In: The 15th IEEE International Conference on Intelligent Transportation Systems, pp. 1014–1019.

Moreira-Matias, L., Gama, J., Ferreira, M., Mendes-Moreira, J., Damas, L., 2013. Predicting taxi–passenger demand using streaming data. IEEE Trans. Intell. Transp. Syst. 14, 1393–1402.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: machine learning in Python. J. Mach. Learn. Res. 12, 2825–2830.

Polson, N.G., Sokolov, V.O., 2017. Deep learning for short-term traffic flow prediction. Transp. Res. Part C: Emerg. Technol. 79, 1–17.

Qian, X., Ukkusuri, S.V., 2015. Spatial variation of the urban taxi ridership using GPS data. Appl. Geography 59, 31–42.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1985. Learning internal representations by error propagation. Technical Report. California Univ San Diego La Jolla Inst for Cognitive Science.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., et al., 1988. Learning representations by back-propagating errors. Cogn. Model. 5, 1.

Shi, X., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.c., 2015. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In: Advances in Neural Information Processing Systems, pp. 802–810.

Soriguera, F., Robusté, F., 2011. Estimation of traffic stream space mean speed from time aggregations of double loop detector data. Transp. Res. Part C: Emerg. Technol. 19, 115–129.

Sun, S., Huang, R., Gao, Y., 2012. Network-scale traffic modeling and forecasting with graphical lasso and neural networks. J. Transp. Eng. 138, 1358–1367.

Sutskever, I., Hinton, G.E., Taylor, G.W., 2009. The recurrent temporal restricted boltzmann machine. In: Advances in Neural Information Processing Systems, pp. 1601–1608.

Toqué, F., Côme, E., El Mahrsi, M.K., Oukhellou, L., 2016. Forecasting dynamic public transport origin-destination matrices with long-short term memory recurrent neural networks. In: The 19th IEEE International Conference on Intelligent Transportation Systems, pp. 1071–1076.

Vlahogianni, E.I., Karlaftis, M.G., Golias, J.C., 2014. Short-term traffic forecasting: where we are and where we're going. Transp. Res. Part C: Emerg. Technol. 43, 3–19.

Wang, J., Deng, W., Guo, Y., 2014. New Bayesian combination method for short-term traffic flow forecasting. Transp. Res. Part C: Emerg. Technol. 43, 79–94.

Wang, J., Gu, Q., Wu, J., Liu, G., Xiong, Z., 2016a. Traffic speed prediction and congestion source exploration: a deep learning method. In: The 16th IEEE International Conference on Data Mining, pp. 499–508.

Wang, J., Shi, Q., 2013. Short-term traffic speed forecasting hybrid model based on chaos–wavelet analysis-support vector machine theory. Transp. Res. Part C: Emerg. Technol. 27, 219–232.

Wang, X., He, F., Yang, H., Gao, H.O., 2016. Pricing strategies for a taxi-hailing platform. Transp. Res. Part E: Logist. Transp. Rev. 93, 212–231.

Wang, X., Jiang, R., Li, L., Lin, Y., Zheng, X., Wang, F.Y., 2017. Capturing car-following behaviors by deep learning. IEEE Trans. Intell. Transp. Syst., in press, http://dx.doi.org/10.1109/TITS.2017.2706963

Wei, Y., Chen, M.C., 2012. Forecasting the short-term metro passenger flow with empirical mode decomposition and neural networks. Transp. Res. Part C: Emerg. Technol. 21, 148–162.

Williams, R.J., Zipser, D., 1989. A learning algorithm for continually running fully recurrent neural networks. Neural Comput. 1, 270–280.

Yang, H., Fung, C., Wong, K., Wong, S.C., 2010a. Nonlinear pricing of taxi services. Transp. Res. Part A: Policy Pract. 44, 337–348.

Yang, H., Leung, C.W., Wong, S.C., Bell, M.G., 2010b. Equilibria of bilateral taxi–customer searching and meeting on networks. Transp. Res. Part B: Methodol. 44, 1067–1083.

Yang, H., Wong, S.C., Wong, K., 2002. Demand–supply equilibrium of taxi services in a network under competition and regulation. Transp. Res. Part B: Methodol. 36, 799–819.

Yang, H., Yang, T., 2011. Equilibrium properties of taxi markets with search frictions. Transp. Res. Part B: Methodol. 45, 696–713.

Yang, H., Ye, M., Tang, W.H.C., Wong, S.C., 2005. A multiperiod dynamic model of taxi services with endogenous service intensity. Oper. Res. 53, 501–515.

Yu, B., Lam, W.H., Tam, M.L., 2011. Bus arrival time prediction at bus stop with multiple routes. Transp. Res. Part C: Emerg. Technol. 19, 1157–1170.

Yu, H., Wu, Z., Wang, S., Wang, Y., Ma, X., 2017. Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks. Sensors 17, 1501.

Yu, R., Abdel-Aty, M.A., Ahmed, M.M., Wang, X., 2014. Utilizing microscopic traffic and weather data to analyze real-time crash patterns in the context of active traffic management. IEEE Trans. Intell. Transp. Syst. 15, 205–213.

Zha, L., Yin, Y., Yang, H., 2016. Economic analysis of ride-sourcing markets. Transp. Res. Part C: Emerg. Technol. 71, 249–266.

Zhang, D., He, T., Lin, S., Munir, S., Stankovic, J.A., 2017a. Taxi-passenger-demand modeling based on big data from a roving sensor network. IEEE Trans. Big Data 3 (3), 362–374.

Zhang, J., Zheng, Y., Qi, D., 2017b. Deep spatio-temporal residual networks for citywide crowd flows prediction. In: AAAI, pp. 1655–1661.

Zhang, N., Zhang, Y., Lu, H., 2011. Seasonal autoregressive integrated moving average and support vector machine models: prediction of short-term traffic flow on freeways. Transp. Res. Rec.: J. Transp. Res. Board 2215, 85–92.

Zhao, K., Khryashchev, D., Freire, J., Silva, C., Vo, H., 2016. Predicting taxi demand at high spatial resolution: approaching the limit of predictability. In: IEEE International Conference on Big Data.

Zhao, Z., Chen, W., Wu, X., Chen, P.C., Liu, J., 2017. Lstm network: a deep learning approach for short-term traffic forecast. IET Intel. Transp. Syst. 11, 68–75.

Zhu, J.Z., Cao, J.X., Zhu, Y., 2014. Traffic volume forecasting based on radial basis function neural network with the consideration of traffic flows at the adjacent intersections. Transp. Res. Part C: Emerg. Technol. 47, 139–154.