Taxi-Based Mobility Demand Formulation and Prediction Using Conditional Generative Adversarial Network-Driven Learning Approaches

Hao Yu, Xiaofeng Chen, Zhenning Li, Guohui Zhang, *Member, IEEE*, Pan Liu, Jinfu Yang, and Yin Yang, *Member, IEEE*

***Abstract*— In this paper, a deep learning (DL) framework was proposed to predict the taxi-passenger demand while the spatial, the temporal, and external dependencies were considered simultaneously. The proposed DL framework combined a modified density-based spatial clustering algorithm with noise (DBSCAN) and a conditional generative adversarial network (CGAN) model. More specifically, the modified DBSCAN model was applied to produce a number of sub-networks considering the spatial correlation of taxi pick-up events in the road network. And the CGAN model, fed with the historical taxi passenger demand and other conditional information, was capable to predict the taxi-passenger demands. The proposed CGAN model was made up with two long short-term memory (LSTM) neural networks, which are termed as the generative network G and the discriminative network D, respectively. Adversarial training process was conducted to the two LSTMs. In the numerical experiment, different model layouts were compared. It was found that different network layouts provided reasonable accuracy. With limited training data, more LSTM layers in the generator network resulted in not only higher accuracy, but also more difficulties in training. Comparisons were also conducted between the proposed prediction model and four typical approaches, including the moving average method, the autoregressive integrated moving method, the neural network model, and the LSTM neural network model. The comparison results showed that the proposed model outperformed all the other methods. And the repeated experiment indicated that the proposed CGAN model provided significant better predictions than the LSTM model did. Future research was recommended to include more datasets for testing the model and more information for improving predictive performance.**

Manuscript received April 22, 2018; revised November 3, 2018 and April 4, 2019; accepted May 31, 2019. Date of publication July 11, 2019; date of current version October 2, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 61803083 and in part by the China Post-Doctoral Science Foundation under Grant 2018M630497. The Associate Editor for this paper was Y. Lv. *(Corresponding author: Guohui Zhang.)*

H. Yu, Z. Li, and G. Zhang are with the Department of Civil and Environmental Engineering, University of Hawaii, Honolulu, HI 96822 USA (e-mail: guohui@hawaii.edu).

X. Chen is with the School of Automation, Northwestern Polytechnical University, Xi'an 710129, China.

P. Liu is with the School of Transportation, Southeast University, Nanjing 210096, China.

J. Yang is with the Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China.

Y. Yang is with the Department of Electrical and Computer Engineering, University of New Mexico, Albuquerque, NM 87131 USA.

Digital Object Identifier 10.1109/TITS.2019.2923964

***Index Terms*— Taxi system, machine learning, demand forecast, big transportation data.**

I. Introduction

TRADITIONALLY, taxi plays an important role in urban transportation systems. Different from other public transport modes, the taxi provides flexible door-to-door service and 24-7 operations [36]. However, with the boom of the mobile Internet, the taxi is slowly losing its market share to various on-demand ride services, such as tailored taxis and lift-sharing cars [5]. By the end of 2015, the number of taxis in Beijing increased by 1.09%, and up to 68,284, while the daily trips were reduced from 1,311,671 to 1,149,726, compared to 2014 [27]. Difference between the traditional and the on-demand services is mainly located on the field of a user's hailing experience. Taxi-passengers had to stand along the road and wait a long time for a taxi, especially during peak hours. It was hard for traditional taxi to handle the random taxi-passenger demands.

A critical challenge for not only the traditional taxi industry but also the on-demand service industry, however, is how to meet the demand with a finite supply. It is realized that having a better understanding of the time-varying taxi-passenger demand over different spatial zones is of great importance to the operator, who can incentivize drivers to be in zones with more potential passenger demands and thus improve the utilization rate of the taxi vehicles [12]. Numerous studies have been conducted to explore the relationship between passenger demands and various causal factors [11], [20], [26], [33], [34]. However, the prediction of taxi-passenger demand is still an open issue and a great challenge mainly affected by three types of dependencies [12], [37]:

- Spatial dependencies: passenger demand was not only determined by variables of current spatial units, but also dependent on the variables from the whole area. As revealed in [35], the variables of nearby spatial units made greater impacts than the distant unit;
- Temporal dependencies: on one hand, just like many other traffic variables, passenger demand varied periodically 25]; on the other hand, time-varying passenger

demand was affected by the past passenger demand in the area.
- External dependencies: external factors, such as weather conditions and traffic regulations, might strongly affect the passenger demand both spatially and temporally.

Recent advances in sensor and wireless communications, such as global positioning system (GPS) and Wi-Fi, have provided a new way to determine vehicle status and location accurately. Most taxi vehicles are now equipped with these types of facilities, producing a source of rich spatial-temporal information. In this paper, the taxi pick-up information was employed in a deep learning (DL) framework, namely a conditional generative adversarial network (CGAN) model, to predict taxi-passenger demand while considering the three dependencies simultaneously. Different from most existing studies, the proposed DL framework focused on mining the spatial-temporal correlation of taxi pick-up events and producing conditional taxi-passenger demand predictions. The main contributions of this paper were concluded as follows:

- The modified density-based spatial clustering algorithm with noise (DBSCAN) algorithm was capable to capture the spatial dependencies among raw taxi pickup information considering traffic road network;
- Long short-term memory (LSTM) structure was used in the proposed CGAN model, which enabled the CGAN model to capture temporal patterns naturally;
- The conditional information is considered in terms of the historical demand information, the temporal information, and the road network information. And the proposed CGAN model is capable of making predictions for time-varying taxi-passenger demand with the conditional information;
- The proposed prediction framework was evaluated using two sets of field data collected in different urban traffic networks. In the numerical experiment, the proposed model outperformed other approaches.

The remainder of the article was structured as follows. After this brief introduction, Section II presented a background review on taxi-passenger demand prediction methods. In Section III, introduction for the modified DBSCAN method and the CGAN model was provided. The prediction framework for taxi demand prediction was then presented in section IV. Section V described the predictive performance of the proposed framework in numerical cases. At the end, we concluded the paper with discussions on future research in Section VI.

II. Background Review

Over the past decades, taxi-passenger demand prediction has attracted the attention of both researchers and companies due to the revolution caused by various taxi-like vehicle servers. Methods of predicting taxi-passenger demand could generally be classified as the classical statistical methods, including those based on causal analysis, the time-series analysis, and the DL methods.

Causal analysis related to taxi-passenger demand prediction includes regression analysis methods and equilibrium model methods [28]. Regression analysis methods usually estimate taxi-passenger demand through the causal relationships among two or more variables [1], [3], [11], [20]. A list of principal variables and parameters considered in the taxi models was summarized in [29]. The principal assumption in these regression analysis methods is the underlying relationship between the passenger demand and the causal factors. Independent sample data is commonly required. Accordingly, the spatial-temporal correlation could not be well handled in those regression analysis models. Equilibrium models tend to explore the demand-supply relationship in the taxi market by assuming a spatial homogeneous or heterogeneous distribution [12], [16], [32]–[35]. However, as indicated in [22], a regional disequilibrium might occur if there exists an excess in waiting passengers in that region, which leads to an imbalanced supply-demand status. Time-series analyses use trends in past observations to forecast future value. Typical time-series methods include moving average (MA) methods, autoregressive integrative moving average (ARIMA) methods [10], and hybrid methods. For instance, [23] proposed a streaming data based sliding-window ensemble framework for taxi-passenger demand prediction. Moreover, three types of time-series models, i.e., the time-varying Poisson model, the weighted time-varying Poisson model, and the ARIMA model, were combined to form a prediction.

Recently, more and more DL approaches have been applied in the field of traffic prediction because they are capable of capturing complex relationships from a huge amount of data [4], [9], [12], [18], [31] proposed two-level architecture, which consisted of a deep belief network at the bottom and a multi-task regression layer at the top, for traffic flow prediction. Reference [18] proposed deep-learning based traffic flow prediction model, i.e., the stacked autoencoder method. Reference [15] described a two-step generative modeling framework, which was capable of learning an activity sequence from a large volume of cell phone data. In order to capture the spatial-temporal correlation, [36] proposed a DL-based approach to predict the pedestrian volume in each region of a city collectively. By fusing the convolutional neural network (CNN) and LSTM techniques, [12] formulated a DL approach to forecast the short-term taxi-passenger demand. Reference [14] extended the original generative adversarial networks (GAN) proposed by [7] to a two-level LSTM neural networks, which was capable to capture the spatial-temporal correlation of network-wide traffic statement estimation.

Inspired by [14], in the proposed prediction framework, the authors combined the modified DBSCAN method and the CGAN model for taxi-passenger demand prediction. Different from previous attempts, such as the CNN models, the recurrent neural network (RNN) models and the GAN model, in this study, the CGAN model applied a conditional adversarial architecture,*i.e.*, a conditional generative network G and a conditional discriminative network D [17], [21]. The conditional term formulated the external dependences. By applying the modified DBSCAN in data processing, adversarial training process in CGAN model helped conditional generator G to capture the spatial, the temporal, and the external dependences of taxi demand simultaneously.

TABLE I
DESCRIPTIONS OF NOTATIONS

Type	Notation	DEFINITION
Constant	K	number of sub-networks/clusters
	T	number of time-steps
	J	number of types of interest points
Variable	$f_d(k,t)$	taxi-passenger demand for spatial-temporal unit (k,t)
	$f_l(k,t)$	link length for spatial-temporal unit (k,t)
	$fi^j(k,t)$	number of interest point type j for spatial-temporal unit (k,t)
	$f_t(k,t)$	time-period type for spatial-temporal unit (k,t)
CGAN	G	generator in CGAN
Parameter	D	discriminator in CGAN
	x	real data
	y	conditional information
	z	randomly generated sample
	$p_z(z)$	prior noise distribution
	$p_{data}(x)$	underlying real data distribution
	f_t	forget gate operator at time-step t
	o_t	output gate operator at time-step t
	i_t	input gate operator at time-step t
	c_t	memory cell vector at time-step t
	l_G	loss value for generator G
	l_D	loss value for discriminator D
DBSCAN	ε	neighborhood radius
Parameter	m_{ps}	minimum required data objects number
	C_k	kth cluster in output results
	H	set of all the data objects
	p_i	attribute vector for ith taxi-passenger pick-up data object
	$l(p_i, p_j)$	target distance between data object i and j
	$N_\varepsilon(p_i)$	set of data objects within ε from data object i
	$\lvert N_\varepsilon(p_i)\rvert$	number of data objects within ε from data object i

III. PRELIMINARIES

In this section, after defining the problem setting for this paper, we introduced the conditional generative adversarial network [21], and the modified DBSCAN algorithm applied to the traffic road network [2], [6]. For convenience, notations used in this paper were summarized in TABLE I.

Definition 1 (spatial-temporal unit): urban traffic network is clustered into K sub-networks with the DBSCAN algorithm. Moreover, taxi pick-up data, seen as representative of taxi-passenger demand, is aggregated in pre-determined time intervals. And kth connected sub-network in tth time interval is termed as the spatial-temporal unit (k,t).

In preparing the training and testing data set, we further defined the following variables for each spatial-temporal unit.

- Taxi-passenger demand $f_d(k,t)$
  In this study, taxi-passenger pick-up events which occur in sub-network k during time-step t are identified as the taxi-passenger demand for spatial-temporal unit (k,t). Furthermore, $f_d(k,t)$ represents the number of taxi-passenger pick-up events.
- Link length $f_l(k,t)$
  Link length for spatial-temporal unit (k,t) is defined as the total length of efficient road links within spatial unit k. In most cases, $f_l(k,t)$ keeps stable among neighboring time intervals. However, real-time traffic regulations may change the availability of links, which modifies link length. In this study, the link length is static information.
- Point of interest $\{fi^j(k,t)\}$
  Point of interest for spatial-temporal unit (k,t) consists of various types of interest points within the spatial unit k. Let J be the total number of interest point types, and $fi^j(k,t)$ is the number of jth of interest point types. It is noted that $\{fi^j(k,t)\}$ varies according to hours of operation.
- Type of time $f_t(k,t)$
  Discrete time-steps are empirically grouped into three types of time periods, i.e., the peak time, off-peak time, and the sleeping time. More specifically, hourly taxi-passenger pick-up counts are ranked from high to low. The first 8 hours are considered peak time. The next 8 hours are off-peak time, and the remaining hours are sleeping time. Then type of time variable $f_t(k,t)$ is given as below:

$$f_t(k,t) = \begin{cases} 0 & \text{peak time} \\ 1 & \text{off-peak time} \\ 2 & \text{sleeping time} \end{cases} \tag{1}$$

Similar approach has also been applied in [12].

In addition, as a common procedure, before training and validating the proposed model, the authors normalized all the variables to $[-1, 1]$ range as below:

$$\theta_*(k,t) = \frac{2f_*(k,t) - f_*^{\min} - f_*^{\max}}{f_*^{\max} - f_*^{\min}} \quad f_* \in \left\{ f_d, f_l, f_i^j, f_t \right\} \tag{2}$$

where $f_*^{\max}$ and $f_*^{\min}$ refer to the maximum and minimum f_* variable in all the spatial-temporal units.

A. Conditional Generative Adverse Networks

Introduced as a novel way to train generative models, GAN has been one of the pioneering deep learning technologies [17]. In the adversarial architecture, the generative network G captures the data distribution, and the discriminative network D estimates the probability that a sample came from the training data rather than G. When both the generator G and the discriminator D are conditioned on some extra information y, the original GAN could be extended to the CGAN model. Here the extra information can be any kind of auxiliary information, such as class labels in image recognition, feature labels in spatial-temporal taxi pick-up data, or data from other modalities. Conditional information y can be fed into both the discriminator D and generator G as an additional input component [21].

As illustrated in Fig.1, to learn a generator distribution p_{data} over data x, the generator G builds a mapping function from a prior noise distribution $p_z(z)$ and conditional information y to data space as $G(z|y)$. Additionally, the discriminator D outputs a scalar $D(x|y)$, which represents the probability that data $x$comes from real data but not the generator G. Simultaneously training process is applied to G and D. The parameters in generator G are adjusted to fool the discriminator D, i.e., to minimize $\log(1\text{-}D(G(z|y)|y))$; and parameters in discriminator D are adjusted to maximize the probability

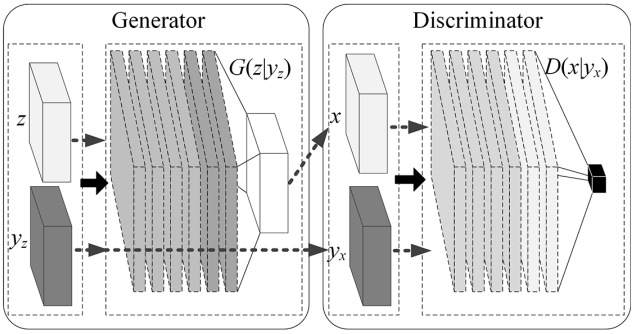

Fig. 1. Typical CGAN architecture used in the proposed structure.

of correct label to both training data and samples from G, i.e., to maximize $\log(D(x|y))$. In other words, D and G play the following two-player minimax game with value function $V(G,D)$:

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}\left[\log D\left(x|y_x\right)\right] + E_{z\sim p_z(z)}\left[\log\left(1-D\left(G\left(z|y_z\right)|y_z\right)\right)\right] \quad (3)$$

B. Modified DBSCAN Algorithm

The density-based spatial clustering algorithm with noise (DBSCAN) is one of the most popular clustering algorithms over the past decades. Compared to other well-known clustering algorithms, DBSCAN is able to discover clusters with arbitrary shapes with minimal domain knowledge required to determine the input parameters [6]. In this study, a modified DBSCAN algorithm was used to capture the spatial correlation in taxi-passenger demand.

Generally speaking, the modified DBSCAN is applied to divide a traffic network into a number of sub-networks according to density-based spatial correlations. Let $p = \{p_1, \ldots, p_m\}$ be the dataset containing m data objects, each of which is described by an attribute vector, p_i. In addition, let L be (conceptually only) an $m \times m$ matrix containing the targeted distance $l(p_i,\ p_j)$ between pairs of objects of x in a metric space. In our modified DBSCAN, the matrix $L$is obtained by calculating the length of the shortest route for any pair of objects. Similar to the original DBSCAN, two important input parameters in the algorithm are the neighborhood radius ε and minimum required data objects number m_{ps}. In the following, we redefined the parameters in a way that made the algorithm more consistent with the proposed DBSCAN method.

Definition 2 (Core, Border and Noise Objects): An object p_i is determined as a core object w.r.t. ε and m_{ps} if its ε-neighborhood contains at least m_{ps} objects, that is, if $|N_\varepsilon(p_i)| \geq m_{ps}$, where $N_\varepsilon(p_i) = \{p_j \in p | l(p_j,\ p_i) \leq \varepsilon\}$ and $|\cdot|$ is the cardinality. An object p_j is determined as a border object w.r.t. ε and m_{ps} if: (1) $p_j \in N_\varepsilon(p_i)$; (2) $|N_\varepsilon(p_i)| \geq m_{ps}$; and (3) $|N_\varepsilon(p_j)| < m_{ps}$. An object p_j is determined as a noise object if it is not a core object or a border object.

Definition 3 (ε- reachable): Two objects p_i and p_j are ε-reachable w.r.t. ε and m_{ps} if $p_j \in N_\varepsilon(p_i)$ and $p_i \in N_\varepsilon(p_j)$.

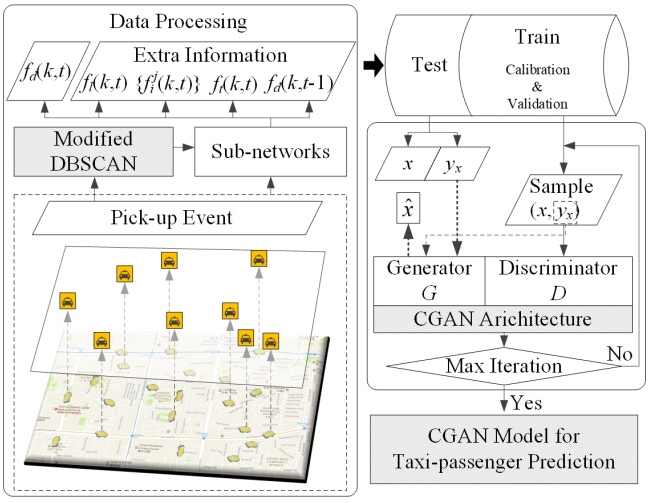

Fig. 2. Framework overview for taxi passenger demand prediction.

Definition 4 (ε- connected): Two core objects p_i and p_j are ε-connected w.r.t. ε and m_{ps} if $N_\varepsilon(p_i) \cap N_\varepsilon(p_j)$ is non-empty.

Definition 5 (Cluster): A cluster C_k w.r.t. ε and m_{ps} is a non-empty maximal subset such that for any core object p_j in C there exists at least one ε-*connected* core object p_i in C, and for any border object p_j in C there exists at least one ε-reachable core object p_i in C.

By calculating the distance matrix, we focused on the spatial correlation via the road network, which was beyond the real network accessibility. In other words, we ignored most of the traffic regulations, such as one-way links and detour signs, which might result in a difference between $l(p_i,\ p_j)$ and $l(p_j,\ p_i)$. Accordingly, the new definitions imply that the modified DBSCAN is formed based on a symmetric notion of reachability. The clustering algorithm for the original DBSCAN is still suitable for this traffic network-based version, except that the distance in the modified DBSCAN is the length of the shortest path connecting the two object events in the traffic network.

IV. Deep Learning Framework for Taxi-Passenger Demand Prediction

In this section, we presented an explicit description of our prediction framework, including data processing using modified DBSCAN algorithm, the architecture of proposed CGAN, and training and prediction procedure. A flow chart for the proposed prediction framework was illustrated in Fig. 2.

A. Data Processing

The raw data includes taxi GPS data, taximeter data, and the road network information. The taxi GPS record has the following attributes: vehicle identification (ID), time, location, speed; and the taximeter record had vehicle ID, pick-up time, and drop-off time. Note that actual vehicle identification, such as the Vehicle Identification Number or license plate number, is not necessary. The vehicle ID here was employed to match data record and to obtain the taxi pickup location.

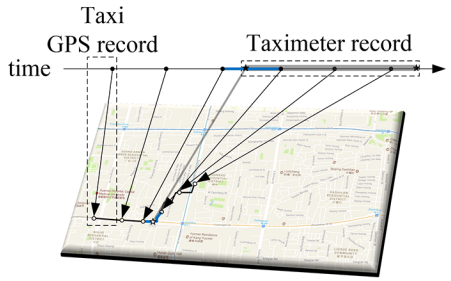

Fig. 3. Path-based linear interpolation located the pick-up event using two steps: firstly, calculate the shortest path between the two GPS records associated with a pick-up time record; secondly, locate the pick-up event on the path linearly according to the time ratio.

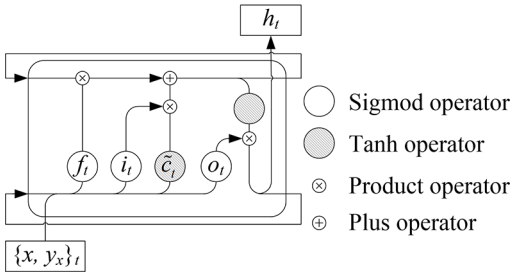

Fig. 4. Illustration of the principle of LSTM cell.

Data masking can be applied to protect privacy information and does not make any difference. Combining the taxi GPS record and taximeter record, we located the pick-up event onto the road network through a path-based linear interpolation. Fig. 3 explained the path-based linear interpolation.

Each pick-up location was processed as a data object. Then we applied the modified DBSCAN algorithm to all the taxi pick-up records. Considering the large amount of pick-up objects, two approaches were jointly applied to eliminate most objects, which reduced computation significantly. Firstly, any object pairs with large difference, i.e., 0.0015 degree, in longitude or latitude were rejected without calculating the actual distance. It was noted that a 0.0015-degree difference in studying area indicated a field distance more than 150m. Secondly, a novel shortest path calculation process was applied to simplify the distance computation. Instead of calculating the shortest path for every object pair, a shortest path matrix connecting the intersections was prepared. Distances between an arbitrary object and its neighboring intersections were calculated using Euclidean distance. The shortest path connecting the two objects was the minimal summation of distance between the taxi object and the intersection, and the shortest path between the two corresponding intersections, which can be obtained from the prepared path matrix.

Density-based pick-up clusters were then labeled to each pick-up record. Furthermore, the same label was also assigned to the corresponding road segment units. Road segment units with the same cluster number formed a sub-network. Let K be the total number of sub-networks. Accordingly, the whole road network was divided into K sub-networks. All the data, including the taxi GPS records, and the road network information, was divided into two parts, i.e., the testing and training datasets, *w.r.t.*, the K sub-networks and the pre-specified time intervals (10 minutes in this study). Moreover, for an arbitrary time interval t, taxi pick-up count and link length were all K-element vectors; the point of interest was actually a $J * K$ matrix; and time of day was one-element vector considering the factor.

B. Proposed CGAN Architecture

Note that the two neural networks, i.e., generator G and discriminator D, could be formulated in any type. In this study, in order to capture the temporal dependencies naturally, two LSTM neural networks were utilized in the proposed CGAN architecture.

LSTM neural networks, introduced as a special RNN architecture [8], are beneficial for dealing with long-term memory in temporal issues [19]. A LSTM cell maps the input vector sequence $\{x,\ y_x\}_t$ to an output vector sequence h_t by T_0 iterations. Fig. 4 depicted the principle of LSTM cell.

The LSTM cell was composed of an input layer, a memory block layer, and an output layer. The memory block contained three types of gating units to control information flow, including the forget gate f_t, input gate i_t, and output gate o_t. And let c_t, $\tilde{c}_t$ be the memory cell vector and the candidate value, respectively. All the information was updated as follows:

$$f_t = \sigma\left(W_f\left[h_{t-1}, \{x, y_x\}_t\right] + b_f\right) \tag{4}$$
$$i_t = \sigma\left(W_i\left[h_{t-1}, \{x, y_x\}_t\right] + b_i\right) \tag{5}$$
$$\tilde{c}_t = \tanh\left(W_c\left[h_{t-1}, \{x, y_x\}_t\right] + b_c\right) \tag{6}$$
$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t \tag{7}$$
$$o_t = \sigma\left(W_o\left[h_{t-1}, \{x, y_x\}_t\right] + b_o\right) \tag{8}$$
$$h_t = o_t * \tanh\left(c_t\right) \tag{9}$$

where function $\sigma(\cdot)$ was the standard logistics sigmoid function defined as follows:

$$\sigma\left(x\right) = \frac{1}{1 + \exp\left(-x\right)} \tag{10}$$

Additionally, function tanh(.) indicated the standard hyperbolic tangent function defined as follows:

$$\tanh\left(x\right) = \frac{\exp\left(x\right) - \exp\left(-x\right)}{\exp\left(x\right) + \exp\left(-x\right)} \tag{11}$$

Based on LSTM cells, a sample layout for both the generator G and discriminator D was presented in Fig. 5. The generator G had three layers, which were one random generation layer and two LSTM layers. The discriminator D had three layers, which were two LSTM layers and one full connected layer. The LSTM layers consisted of multiple LSTM cells described before. In this study, the number of LSTM cells in a single layer was 256. Models with different numbers of LSTM layers in generator G or discriminator D were also tested and presented in the experimental results.

We designed the input of the discriminator D such that it contained a sequence of taxi-passenger demand $x_t = \{\theta_d(k, t)\}_k$ for each sub-network and the corresponding

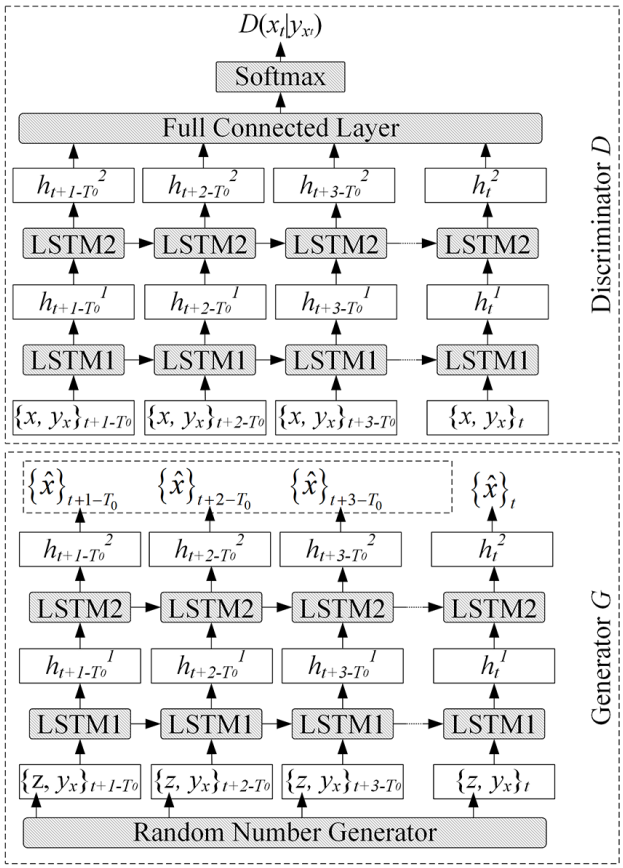

Fig. 5. Layout for the proposed CGAN.

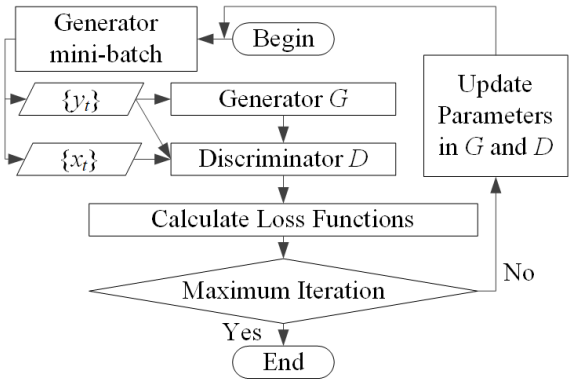

Fig. 6. Training procedure flow chart.

conditional information y_{xt} with multiple time-steps. In this study, the time-step for look-back time window equaled 6. In conditional information, we included the link length, the point of interest information, type of time, and the taxi-passenger demand in previous time step, i.e., y_{xt} is represented as below:

$$y_{x_t} = \left\{ \theta_l(k,t), \left\{ \theta_i^j(k,t) \right\}_j, \theta_t(k,t), \theta_d(k,t-1) \right\}_k \quad (12)$$

With the input information, the output of each LSTM cell in each time-step was a list of value between -1 and 1. The full connected layer together with the softmax function mapped the output of the second LSTM layer to a value between 0 and 1, representing the probability that the input taxi-passenger demand was field-collected data. As for the generator G, in the generation layer, a sequence randomly samples were generated from a uniform distribution scaled between -1 and 1. The size of the random generated sample was just the same as vector $\{x_t\}$. In addition, the uniform distributed random samples, together with the corresponding conditional information, were imported into the two LSTM layers to generate the predictions of taxi-passenger demand. The generative ability of generator G was improved in the adversarial training.

C. Training and Predicting Procedure

The training procedure for the conditional generator G and the conditional discriminator D was illustrated in Fig. 6. It was noted that D and G played a minimax game with a value function defined in EQ (3). Instead of minimizing likelihood of discriminator being correct for generator G, it worked much better to maximize the probability of the discriminator making the incorrect choice [7], [30]. Accordingly, in updating the generator G and discriminator D, we minimized the following loss functions for G and D, respectively.

$$l_G = E_{z\sim p_z(z)}\left[\log D\left(G\left(z\,|y_z\right)|y_z\right)\right] \quad (13)$$

$$l_D = E_{h\sim p_{data}(h)}\left[\log D\left(h\,|y_h\right)\right] + E_{z\sim p_z(z)}\left[\log\left(1 - D\left(G\left(z\,|y_z\right)|y_z\right)\right)\right] \quad (14)$$

As shown in Fig. 6, training process started from sampling mini-batch of examples from database. In this study the batch size was set to 36 as a tradeoff between training time per epoch and quality of model, as well as the required memory size [13]. Each example in the mini-batch consisted of field-collected taxi pick-up set $\{x_t\}$ and conditional information set $\{y_t\}$. The discriminator D was pre-trained using the two sets with an un-trained generator G for certain iterations. The pre-training process only updated the parameters in discriminator D following EQ(14). Then the conditional information $\{y_t\}$ was imported to both the generator G and discriminator D, while $\{x_t\}$ was processed to discriminator. The generator G and discriminator D were updated simultaneously. If the pre-specified maximum iteration was not satisfied, parameters in G and D were updated following EQ (13) and EQ (14); otherwise, stop training.

To prevent overfitting, we further applied the dropout method with probability of 0.6 at each LSTM layer. The number of training epochs was set to 400. It should be noted that convergence on training dataset and prediction performance on testing dataset were different. In this study, we focused on the prediction performance, and we simply trained the model as long as necessary, which ensured the CGAN model captured all the possible spatial-temporal patterns hiding in the training dataset.

After the CGAN was well trained, taxi-passenger demands were generated by the generator G with corresponding conditional information. The predicted taxi pick-up counts were then obtained by applying the following decoding function to

the model outputs of the latest time-step:

$$f_d(k,t) = \frac{\hat{\theta}_d(k,t)}{2}\left(f_d^{\max} - f_d^{\min}\right) + \frac{1}{2}\left(f_d^{\min} + f_d^{\max}\right) \quad (15)$$

It was noted that multiple samples were generated and accordingly, multiple demand predictions were calculated. The average value of the multiple demand predictions was used as the predicted taxi-passenger demand.

V. Experimental Results

This section evaluated the CGAN based taxi-passenger demand prediction model using field collected taxi-passenger pickup data. A two-week data collected in Beijing, China was applied in evaluation tests with different model layouts and performance comparison. And a two-year data collected in New York city was then applied to further validate transferability of the comparison results with larger area and longer duration.

A. Data Description

The studying area locates in downtown Beijing, China, located from 116.09° E to 116.39° E in longitude, and from 39.93° N to 40.23° N in latitude. The taxi GPS data set and taximeter data set used in this paper was provided by Beijing taxi companies. The number of referred taxis was 68,310, and the total number of pickup records in studying area was 1,750,198. As reported in annual city report of Beijing, the total number of operating taxis was around 68,000, which indicated the fact that the authors obtained almost the accessible taxi data during the studying period. The corresponding traffic network data was collected from Open Street Map, while the POI information was provided by Baidu Map.

The GPS data was generated over a period of two weeks (from Nov. 1st, 2015 to Nov. 14th, 2015). More specifically, data from Nov. 1st to Nov. 12th was used as the training dataset and the left two days were used for testing. Within the training dataset, the first 11-day data was used for training, while 12th day's data was used for validation. It should be noticed that a two-week dataset was not large enough to cover various traffic patterns/situations, such as traffic accidents, weekend mobility patterns, or adverse weather conditions. The small data case mainly focused on validating the performance of the proposed prediction framework.

The spatial distributions for various POIs and taxi-passenger demand were illustrated in Fig. 7. In each sub-figure, the shade of color represented the density of different parameters. Moreover, Fig. 8 depicted time-varying taxi-passenger demand throughout a whole day. It was observed that the peak hours for taxi-passenger demand occur in two periods, from 8:00 am to 14:00 pm and 16:00 pm to 18:00 pm. This finding was slightly different from the peak-hours of urban traffic volume, which usually occurred during commuting time, i.e., from 7:00 am to 10:00 am and from 16:30 pm to 19:30 pm. According to field-collected data, the sleeping time is from 23:00 pm to 7:00 am and the remaining time is considered off-peak time.

In order to apply the modified DBSCAN algorithm, the neighborhood radius ε was set as 100 m, and the minimum required data objects number m_{ps} was set as 80 to balance the taxi-passenger demand among different clusters. It was clear that the value of the two parameters affected the clustering result significantly, and influences the prediction performance eventually. Here, the parameters were roughly determined according to their physical meanings. More specifically, by applying the 100m neighborhood radius, it was

Fig. 7. Distribution of various POI points as well as the taxi passenger demand.

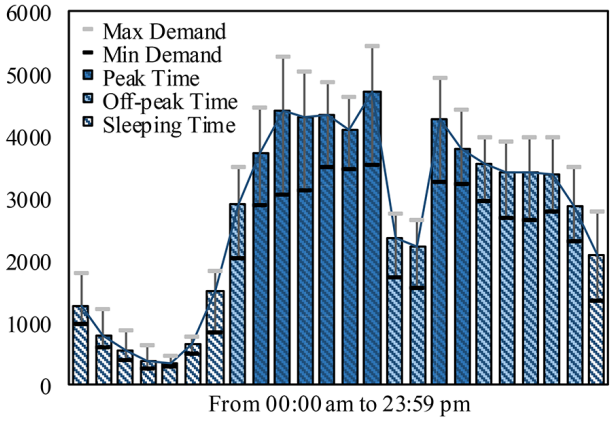

Fig. 8. Taxi-passenger demands vary with time.

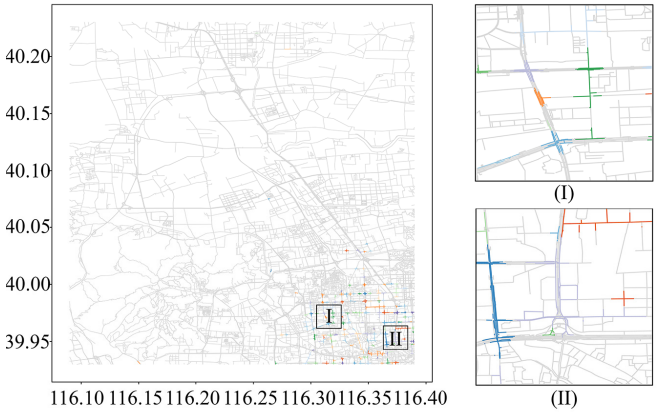

Fig. 9. Cluster result of the taxi-passenger demand according to the DBSCAN algorithm.

assumed that the radius of taxi-passenger searching behavior was 100 m. It would be of great interesting to exploring the impacts of those parameters on the demand prediction performance in future study.

By applying the modified DBSCAN algorithm to taxi pick-up data, we obtained 159 taxi-passenger demand clusters, as shown in Fig. 9. Since the shortest route information was considered, it was noted that objects were clustered following the road network design, but not the crow-fly distance. Most of the clusters were single or continuous intersections, as shown in Fig. 9 (I) and (II), which was definitely reasonable. Based on the cluster results, 916,025 taxi-passenger pick-up records, which comprised 52.34% of all the demand, entered the 159 clusters, which took only 3.90% of the whole network length.

B. Performance Evaluation for CGAN Based Taxi Demand Prediction

Three performance indexes were used in this study, including the mean squared error (MSE), the mean absolute error (MAE), and the mean absolute percentage error (MAPE). They were calculated as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{KT}\sum_{k\in K}\sum_{t\in T}\left(\hat{c}_p\left(k,t\right)-c_p\left(k,t\right)\right)^2} \tag{16}$$

$$\mathrm{MAE} = \frac{1}{KT}\sum_{k\in K}\sum_{t\in T}\left|\hat{c}_p\left(k,t\right)-c_p\left(k,t\right)\right| \tag{17}$$

$$\mathrm{MAPE} = \frac{1}{KT}\sum_{k\in K}\sum_{t\in T}\frac{\left|\hat{c}_p\left(k,t\right)-c_p\left(k,t\right)\right|}{c_p\left(k,t\right)} \tag{18}$$

where MSE and MAE were applied to all the test dataset, and MAPE was applied to the spatial-temporal unit with the top 5% of taxi-passenger demand. In Beijing dataset, the top 5% of spatiotemporal units referred to units with no less than 11 pick-up records.

Considering the pre-specified batch size of 36 and epoch size of 400, it was 44 iterations per epoch, and 17600 iterations in total. Validation test results with different model layouts, *i.e.*, well trained CGAN models with different numbers of

TABLE II
TEST RESULTS WITH DIFFERENT MODEL LAYOUTS

Number of LSTM Layers in D	Number of LSTM Layers in G: 1	2	3
1	MSE: 1.257 MAE: 0.772 MAPE[A]: 0.152	MSE: 1.110 MAE: 0.697 MAPE[A]: 0.131	MSE: 1.064 MAE: 0.695 MAPE[A]: 0.131
2	MSE: 1.130 MAE: 0.728 MAPE[A]: 0.150	MSE: 0.977 MAE: 0.628 MAPE[A]: 0.130	MSE: 1.003 MAE: 0.657 MAPE[A]: 0.132
3	MSE: 1.112 MAE: 0.708 MAPE[A]: 0.143	MSE: 1.006 MAE: 0.642 MAPE[A]: 0.132	MSE: 1.010 MAE: 0.688 MAPE[A]: 0.132

[A]MAPE INDEX WAS APPLIED TO THE SPATIAL-TEMPORAL UNIT WITH MORE THAN ELEVEN PICK-UP RECORDS.

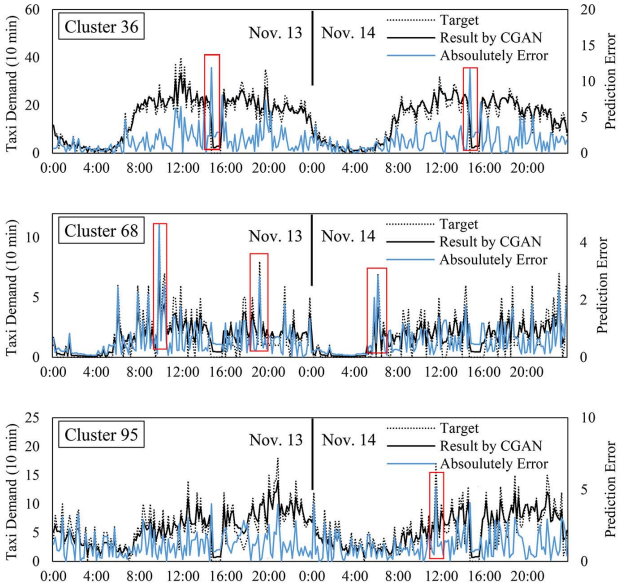

Fig. 10. Taxi demand prediction results in three sub-networks using CGAN model.

LSTM layer, were listed in TABLE II. It was clear that with the increasing LSTM layers in G, the CGAN prediction models presented better performance. Moreover, it was found that with only one LSTM layer in G, the CGAN models performed worse than the others. When the numbers of LSTM layer in G were two or three, the performance difference among these models were much closer. This phenomenon may be the result of limited data. It would be more difficult to train a deeper model well. On the other hand, the test result also showed that the impact of LSTM layer number in D is a little complex. When the LSTM layer number in D increased from one to two, models performed better in all of the three indexes. But when the LSTM layer number in D increased from two to three, the models provide a little better or similar performance in some indexes, but worse performance in the others. The two possible explanations were: firstly, as mentioned, the limited data made it harder to train a deeper model well; secondly, more LSTM layers in D resulted in the prematurity of the Adam Optimizer applied in this study.

Fig. 10 plotted the predicted results for three random selected clusters (sub-networks) with two LSTM layers in

both G and D. It showed that the proposed CGAN model provided reasonably accurate forecasts of the taxi-passenger demand. In general, the proposed model had higher prediction errors for greater taxi demand and sudden-drop situations, as shown in the red rectangle marks. Except for the limitation on dataset size, the limited conditional information involved in this study made it difficult to capture those atypical conditions. Future research would involve additional information, including the real-time traffic volume, weather conditions, and possible traffic events, to predict those outliers more accurately.

C. Model Comparisons

The well-trained model was compared with six other prediction methods, including the moving average (MA) method, the autoregressive integrated moving average (ARIMA) method, the neural network (NN) model, the convolutional neural network (CNN) model, the LSTM neural network, and the CNN-LSTM model. They were briefly described as follows:

(1) MA: the MA method has been widely used in time-series analysis, which calculates predictions using the mean value of nearest historical records. In this study, the time window equaled three time-steps, i.e., the taxi-passenger counts in the spatial-temporal unit $(k,t\text{-}3)$, $(k,t\text{-}2)$, and $(k,t\text{-}1)$ were used to predict the taxi-passenger counts in the spatial-temporal unit (k,t).

(2) ARIMA: the ARIMA model was introduced by Box and Jenkins (1976) [38]. In an ARIMA $(p,\ d,\ q)$ model, the parameter p and q are integers and referred to as the orders of the autoregressive term and moving average term. The parameter d represents the dth order difference from the original data series, which aims to remove the trend from the data series. In this study, parameters were optimized using the auto-optimal function in forecast package with R-3.4.4.

(3) NN: the NN model [24] applied in this study involved two hidden layer with 256 hidden units and one full connected layer. It was fitted with multiple random starting weights. The conditional information $\{y_t\}$ was used as an input and the output was the scaled taxi-passenger pick-up $f_p(k,t)$ for each spatial-temporal unit (k,t).

(4) CNN: the CNN model applied in this study employed two one-dimension convolutional layers and a full connection layer. The conditional information $\{y_t\}$ was imported as a picture with one-dimension and one channel.

(5) LSTM: the LSTM neural network was also applied for taxi demand prediction. The LSTM model evaluated here captured the same structure as the generative network G in the proposed CGAN model, i.e., two LSTM layers and one full connected layer. More specifically, conditional information $\{y_t\}$ was used as an input considering 6 time-steps for look-back time window, while the output was the scaled taxi-passenger pick-ups.

(6) CNN-LSTM: a CNN-LSTM model is an integration of a CNN part and an LSTM part, which captures the spatial and temporal dependencies simultaneously. The CNN-LSTM model employed here consist of one CNN layer, one LSTM layer and a full connection layer. The look-back time window is set to 6. Conditional information $\{y_t\}$ of the 6 time-steps was imported to the CNN layer and then the LSTM layer. The output was the scaled taxi-passenger pick-ups in the following time-step.

TABLE III
COMPARISON RESULTS AMONG CGAN MODEL AND OTHER TYPICAL MODELS

MODEL		MSE	MAE	MAPE[A]
MA	MIN	0.539	0.350	0.000
	MAX	7.136	5.075	0.803
	MEAN	1.811	1.348	0.146
	STD.	1.058	0.783	0.238
	TOTAL[B]	2.096	1.348	0.316
ARIMA	MIN	0.514	0.405	0.000
	MAX	7.629	5.283	0.846
	MEAN	1.530	1.336	0.149
	STD.	1.153	0.800	0.251
	TOTAL[B]	1.736	1.336	0.306
NN	MIN	0.329	0.265	0.000
	MAX	5.353	3.846	0.583
	MEAN	1.511	1.146	0.097
	STD.	0.785	0.589	0.177
	TOTAL[B]	1.701	1.146	0.173
CNN	MIN	0.346	0.294	0.000
	MAX	4.777	3.314	0.564
	MEAN	1.343	0.969	0.091
	STD.	0.542	0.675	0.126
	TOTAL[B]	1.448	0.969	0.157
LSTM	MIN	0.288	0.196	0.000
	MAX	3.449	2.208	0.443
	MEAN	1.068	0.680	0.076
	STD	0.427	0.329	0.126
	TOTAL[B]	1.130	0.680	0.151
CNN-LSTM	MIN	0.176	0.196	0
	MAX	3.551	2.321	0.636
	MEAN	1.002	0.678	0.75
	STD	0.546	0.398	0.131
	TOTAL[B]	1.072	0.678	0.149
CGAN	MIN	0.259	0.193	0.000
	MAX	3.157	2.413	0.405
	MEAN	0.877	0.655	0.069
	STD	0.524	0.391	0.115
	TOTAL[B]	0.945	0.655	0.138

[A]MAPE INDEX WAS APPLIED TO THE SPATIAL-TEMPORAL UNIT WITH MORE THAN ELEVEN PICK-UP RECORDS.
[B]HERE TOTAL MEANS THAT THE INDEX WAS APPLIED TO ALL THE SUB-NETWORKS.

It should be noted that here the MA, and the ARIMA were calibrated for each single subnetwork and the predictions for each subnetwork were obtained separately, while the rest models were applied to the whole network. TABLE III presented the comparison results of the proposed CGAN model, as well as the MA, the ARIMA, the NN, the CNN, the LSTM, and the CNN-LSTM approaches. Note that, here the CGAN prediction model referred the CGAN layout with two LSTM layers in both G and D. As shown in TABLE III, each index was calculated for each sub-network, respectively. It was found that the proposed CGAN model outperformed the six typical approaches in all the three performance indexes, as expected, followed by the CNN-LSTM, the LSTM, the CNN, the NN, the ARIMA, and the MA.

It was of interest to take a further look at the five neural network-based models. Compared to the basic NN model,

TABLE IV
RESULTS OF TRANSFERABILITY TEST

MODEL		MSE	MAE	MAPE[A]
MA	MIN	2.294	0.615	0.000
	MAX	58.612	23.329	2.843
	MEAN	16.653	6.541	0.211
	STD.	7.034	2.383	0.202
	TOTAL[B]	17.955	6.541	0.220
ARIMA	MIN	2.234	1.102	0.000
	MAX	56.232	24.432	2.258
	MEAN	15.645	6.133	0.210
	STD.	11.790	3.184	0.203
	TOTAL[B]	17.425	6.133	0.219
NN	MIN	2.976	1.309	0.000
	MAX	58.487	26.541	2.227
	MEAN	15.426	6.089	0.213
	STD.	7.676	3.030	0.201
	TOTAL[B]	17.230	6.089	0.230
CNN	MIN	2.637	1.259	0.000
	MAX	45.861	17.131	2.351
	MEAN	15.210	6.067	0.204
	STD.	7.689	3.045	0.205
	TOTAL[B]	17.043	6.067	0.216
LSTM	MIN	1.797	0.856	0.000
	MAX	54.778	21.492	1.712
	MEAN	10.946	4.332	0.182
	STD	5.538	2.106	0.196
	TOTAL[B]	12.275	4.332	0.186
CNN-LSTM	MIN	1.910	0.883	0.000
	MAX	47.371	20.938	1.767
	MEAN	10.749	4.275	0.178
	STD	5.147	1.968	0.184
	TOTAL[B]	11.917	4.275	0.186
CGAN	MIN	2.532	1.650	0.000
	MAX	38.582	17.031	0.870
	MEAN	7.654	4.093	0.144
	STD	2.466	1.090	0.118
	TOTAL[B]	8.042	4.093	0.154

[A]MAPE INDEX WAS APPLIED TO THE SPATIAL-TEMPORAL UNIT WITH MORE THAN EIGHTY THREE PICK-UP RECORDS.
[b]HERE TOTAL MEANS THAT THE INDEX WAS APPLIED TO ALL THE SUB-NETWORKS.

the CNN captures the spatial hierarchies of features through backpropagation by using multiple convolution layers, while the LSTM model mainly captures the temporal dependencies. However, it was found that the LSTM improved the MSE by 34.57%, while the CNN model improved the MSE by 14.87%. It may because that the spatial dependencies have partially been considered in the data processing by the modified DBSCAN algorithm. According, the additional temporal dependencies provided more improvements in LSTM. The CNN-LSTM, as a combination of CNN and LSTM models, slightly improved the performance of LSTM model by 5.13% in terms of the MSE index. Compared to the LSTM mode, the proposed CGAN model reduced the MSE, the MAE, and the MAPE for the top 5% spatial-temporal units by 16.36%, 3.68%, and 8.61%, respectively, by training the adversarial architecture.

D. Transferability of Comparison Results

Two-year taxi pickup data from New York city was selected to test the effectiveness of the CGAN model and examine the transferability based on the comparison results. From 2013 to 2014, a total of 350,441,563 taxi pickup records were stored in the dataset, including the geocoded pickup location information and the time instant stamps. The corresponding traffic network data was also collected from Open Street Map, while the POI information was obtained through the internet. The proposed model layout kept unchanged. Among the two-year data, the first 17 month-data was used as the training dataset while the left 7 month-data was used as the testing dataset. The proposed framework was then applied. In summary, a total of 263 sub-networks were obtained via the modified DBSCAN algorithm. And then the predictive performance of the proposed CGAN model and the other approaches were compared in Table IV.

It was found that the comparison results were quite stable. In most of the tests, the rank in the predictive performance of various models either remained the same, or only changed slightly. Once again, the proposed CGAN model outperformed the other approaches in all the three indexes. Besides, the results confirmed the findings that were obtained using the data from Beijing, China. Compared to the NN model, the LSTM model provided higher improvement than the CNN model. And the adversarial training process enhanced the capability of the LSTM model, i.e., the generator G, to capture the spatiotemporal correlation among taxi pickup data.

VI. CONCLUSION AND DISCUSSION

In this study, a DL framework, which combined the modified DBSCAN model and a CGAN model, was proposed to predict the taxi-passenger demand. The spatial dependencies, the temporal dependencies, and the external dependencies were considered in the proposed framework simultaneously. More specifically, the modified DBSCAN model was applied to produce a number of sub-networks according to density-based spatial correlation among taxi pick-up events. Taxi-passenger demand was thus aggregated and predicted based on the sub-networks. And by adversarial training the two multi-layer LSTM networks, the proposed CGAN model was fed with historical taxi-passenger demand information, road network geometric information, land-use information, and time labels. Different model layouts, i.e., different numbers of LSTM layers in generator G and discriminator D, were tested using the data collected in the Beijing area. It was found that all those models provide reasonable predictions. With the limited training data, more layers resulted in more difficulties in training and more layers in the discriminator network did not perform well. Additionally, the proposed CGAN prediction model was compared with six typical approaches, i.e., the MA, the ARIMA, the NN, the CNN, the LSTM, and the CNN-LSTM model. The comparison results indicated that the proposed model outperformed the other approaches in all the three predictive performance indexes. And the proposed CGAN model provided reasonable stability in the transferability test.

The DL framework has the potential to be used for real-time taxi-passenger demand prediction for both the traditional taxi industry and the on-demand vehicle service. Once the model is well trained, the input data involving the past taxi pick-up information and the conditional information can be processed

easily to generate the taxi-passenger demand predictions. Accurate taxi-passenger demand forecasting can thus provide suggestions for these platforms to arrange for the cruising cars to meet the passenger demand. However, before the proposed model is used in practical applications, more research is still needed to further improve the model predictive performance. Firstly, the proposed model currently involves only the taxi pick-up information and some other static information. For future research, we recommend including more information in the CGAN model. For instance, a comprehensive model with both the pick-up and drop-off information would be pursued to improve the predictive accuracy. In that case, those hyper-parameters shall also be investigated. Secondly, optimal algorithm to train the adversarial architecture efficiently is worth consideration. Finally, the interactions among the cluster algorithm, i.e., the modified DBSCAN, the pre-specified time interval, and the proposed prediction model are of great interest and could be explored in future research.

REFERENCES

[1] M. Beesley and S. Glaister, "Information for regulating: The case of taxis," *Econ. J.*, vol. 93, no. 371, pp. 594–615, Sep. 1983.

[2] R. J. G. B. Campello, D. Moulavi, A. Zimek, and J. Sander, "Hierarchical density estimates for data clustering, visualization, and outlier detection," *ACM Trans. Knowl. Discovery Data*, vol. 10, no. 1, pp. 5:1–5:51, 2015.

[3] G. W. Douglas, "Price regulation and optimal service standards: The taxicab industry," *J. Transp. Econ. Policy*, vol. 6, no. 2, pp. 116–127, 1972.

[4] Y. Duan, Y. Lv, Y.-L. Liu, and F.-Y. Wang, "An efficient realization of deep learning for traffic data imputation," *Transp. Res. C, Emerg. Technol.*, vol. 72, pp. 168–181, Nov. 2016.

[5] M. P. Enoch, "How a rapid modal convergence into a universal automated taxi service could be the future for local passenger transport," *Technol. Anal. Strategic Manage.*, vol. 27, no. 8, pp. 910–924, 2015.

[6] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," *KDD*, vol. 96, no. 34, pp. 226–231, 1996.

[7] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[8] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[9] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: Deep belief networks with multitask learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 2191–2201, Oct. 2014.

[10] A. Kaltenbrunner, R. Meza, J. Grivolla, J. Codina, and R. Banchs, "Urban cycles and mobility patterns: Exploring and predicting trends in a bicycle-based public transport system," *Pervas. Mobile Comput.*, vol. 6, no. 4, pp. 455–466, Aug. 2010.

[11] A. Kanafani, *Transportation Demand Analysis*. New York, NY, USA: McGraw-Hill, 1983.

[12] J. Ke, H. Zheng, H. Yang, and X. Chen, "Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach," *Transp. Res. C, Emerg. Technol.*, vol. 85, pp. 591–608, Dec. 2017.

[13] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang, "On large-batch training for deep learning: Generalization gap and sharp minima," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–6.

[14] Y. Liang, Z. Cui, Y. Tian, H. Chen, and Y. Wang, "A deep generative adversarial architecture for network-wide spatial-temporal traffic-state estimation," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2672, no. 45, pp. 87–105, 2018.

[15] Z. Lin, M. Yin, S. Feygin, M. Sheehan, J.-F. Paiement, and A. Pozdnoukhov, "Deep generative models of urban mobility," in *Proc. ACM SIGKDD Conf.*, Aug. 2017, pp. 1–8.

[16] J. Long, W. Szeto, J. Du, and W. Wong, "A dynamic taxi traffic assignment model: A two-level continuum transportation system approach," *Transp. Res. B, Methodol.*, vol. 100, pp. 222–254, Jun. 2017.

[17] Y. Lv, Y. Chen, L. Li, and F.-Y. Wang, "Generative adversarial networks for parallel transportation systems," *IEEE Intell. Transp. Syst. Mag.*, vol. 10, no. 3, pp. 4–10, Jun. 2018.

[18] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.

[19] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," *Transp. Res. C, Emerg. Technol.*, vol. 54, pp. 187–197, May 2015.

[20] M. G. McNally, "The four step model," Center Activity Syst. Anal., UC Irvine, Irvine, CA, USA, Tech. Rep. UCI-ITS-AS-WP-07-2, 2008.

[21] M. Mirza and S. Osindero, "Conditional generative adversarial nets," Nov. 2014, *arXiv:1411.1784*. [Online]. Available: https://arxiv.org/abs/1411.1784

[22] L. Moreira-Matias, J. Gama, M. Ferreira, and L. Damas, "A predictive model for the passenger demand on a taxi network," in *Proc. 15th IEEE Int. Conf. Intell. Transp. Syst.*, Sep. 2012, pp. 1014–1019.

[23] L. Moreira-Matias, J. Gama, M. Ferreira, J. Mendes-Moreira, and L. Damas, "Predicting taxi–passenger demand using streaming data," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1393–1402, Sep. 2013.

[24] K. Ord, S. R. Fildes, and N. Kourentzes, *Principles of Business Forecasting*, vol. 10, 2nd ed. New York, NY, USA: Wessex Press, 2017.

[25] X. Qian and S. V. Ukkusuri, "Spatial variation of the urban taxi ridership using GPS data," *Appl. Geography*, vol. 59, pp. 31–42, May 2015.

[26] X. Qian and S. V. Ukkusuri, "Time-of-day pricing in taxi markets," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 6, pp. 1610–1622, Jun. 2017.

[27] Y. Quan *et al.*, "Annual report of transportation development in Beijing 2016," Beijing Transp. Inst., Beijing, China, Tech. Rep. BTDRI-152016, 2016.

[28] J. M. Salanova, M. Estrada, G. Aifadopoulou, and E. Mitsakis, "A review of the modeling of taxi services," *Procedia-Social Behav. Sci.*, vol. 20, pp. 150–161, Jan. 2011.

[29] J. M. Salanova, M. E. Romeu, and C. Amat, "Aggregated modeling of urban taxi services," *Procedia-Social Behav. Sci.*, vol. 160, pp. 352–361, Dec. 2014.

[30] T. Salimans *et al.*, "Improved techniques for training GANs," *Adv. Neural Inf. Process. Syst.*, 2016, pp. 2234–2242.

[31] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: Where we are and where we're going," *Transp. Res. C, Emerg. Technol.*, vol. 43, pp. 3–19, Jun. 2014.

[32] H. Yang, Y. W. Lau, S. C. Wong, and H. Lo, "A macroscopic taxi model for passenger demand, taxi utilization and level of services," *Transportation*, vol. 27, pp. 317–340, Jun. 2000.

[33] H. Yang, S. C. Wong, and K. I. Wong, "Demand–supply equilibrium of taxi services in a network under competition and regulation," *Transp. Res. B, Methodol.*, vol. 36, no. 9, pp. 799–819, Nov. 2002.

[34] H. Yang, M. Ye, W. H.-C. Tang, and S. C. Wong, "A multiperiod dynamic model of taxi services with endogenous service intensity," *Oper. Res.*, vol. 53, no. 3, pp. 501–515, 2005.

[35] H. Yang, C. W. Y. Leung, S. C. Wong, and M. G. H. Bell, "Equilibria of bilateral taxi–customer searching and meeting on networks," *Transp. Res. B, Methodol.*, vol. 44, pp. 1067–1083, Sep./Nov. 2010.

[36] Y. Zhang, H. Guo, C. Li, W. Wang, X. Jiang, and Y. Liu, "Which one is more attractive to traveler, taxi or tailored taxi? An empirical study in China," *Procedia Eng.*, vol. 137, pp. 867–875, Jan. 2016.

[37] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. 31st AAAI Conf. Artif. Intell.*, Feb. 2017, pp. 1655–1661.

[38] G. E. P. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*. San Francisco, CA, USA: Holden-Day, 1976.

Hao Yu was born in China, in 1988. He received the B.S. and Ph.D. degrees from the School of Transportation, Southeast University, China, in 2010 and 2017, respectively.

He has been a Post-Doctoral Fellow with the Department of Civil and Environmental Engineering, University of Hawaii at Mānoa, since 2017. His research interests include large-scale traffic network modeling, big data on urban traffic management, highway safety analysis, and their practical engineering applications.

Xiaofeng Chen received the Ph.D. degree in control theory and engineering from Northwestern Polytechnical University, China, in 2003. He is currently an Associate Professor with the School of Automation, Northwestern Polytechnical University. His research interests include traffic system modeling and control, traffic detection systems, and embedded system development and related applications.

Zhenning Li received the B.S. and master's degrees from the Harbin Institute of Technology, Harbin, China. He is currently pursuing the Ph.D. degree in civil engineering with the Department of Civil and Environmental Engineering, University of Hawaii, Mānoa. His research interests include transportation system operations and analysis, traffic safety, and connected vehicles.

Guohui Zhang received the Ph.D. degree from the University of Washington (UW), Seattle, in 2000. He is currently an Associate Professor with the Department of Civil and Environmental Engineering, University of Hawaii at Mānoa, Honolulu. His research interests include transportation system operations and analysis, transportation data management and analysis, large-scale traffic simulation, geographic information system-based infrastructure asset management, and traffic detection systems. He serves as a member of the Transportation Research Board (TRB) Committee on Information Systems and Technology (ABJ50) and a Panel Member of five NCHRP projects.

Pan Liu received the Ph.D. degree in civil engineering from the University of South Florida, Tampa, in 2006. He is currently a Professor with the Key Laboratory of Traffic Planning and Management, School of Transportation, Southeast University, Nanjing, China. His research interests include traffic design, traffic safety, and intelligent transportation systems. He was a recipient of the Outstanding Youth Science Foundation of NSFC in 2013 and the Award of the Program for New Century Excellent Talents funded by the Ministry of Education of the People's Republic of China in 2011.

Jinfu Yang received the Ph.D. degree in pattern recognition and intelligent systems from the National Laboratory of Pattern Recognition, Chinese Academy of Sciences, in 2006. He is currently a Professor with the Faculty of Information Technology, Beijing University of Technology, Beijing, China. His research interests include pattern recognition, computer vision, robot navigation, and intelligent transportation systems. He was a recipient of the award of the Importation and Development of High-Caliber Talents Project of Beijing Municipal Institutions.

Yin Yang received the Ph.D. degree in computer science from the University of Texas at Dallas in 2013. He is currently an Assistant Professor with the Department of Electrical Computer Engineering, The University of New Mexico, Albuquerque. His research interests include scientific visualization, physics-based animation/simulation and related applications, and medical imaging analysis.