Region-level Ride-hailing Demand Prediction with Deep Learning

Ruohui Lan
Computer Science, Northwestern Polytechnical University, Xi’an, 710072, China

Abstract. The exponential growth of ride-hailing demand has disrupted the urban transportation and changing the way people travel. To alleviate the shortage problem, many researchers have done numerous experiments to predict the region-level ride-hailing demand which can arrange the vehicles more efficiently. In this paper, we use the online ride-hailing order dataset of Haikou, Hainan Province, China, released by Didi Chuxing. We apply historical models, MLP, CNN and ConvLSTM model for region-level ride-hailing demand prediction in the next hour. Compared to the traditional deep learning models, ConvLSTM has a better performance with the lowest RMSE (4.796) when the input historical length is 24 hours.

1. Introduction
Urban transportation is the basis for the existence and development of a modern city and an important guarantee for the normal operation, until 2018 the number of total passengers of public transport reached 84.841 billion [1]. Taking Beijing as an example, until 2017, average one-way commute distance had reached 11km [2] and it raised to 13km in 2019 [3]. In 2019, more than half of the commuters’ commute distances are higher than 10km. However, traffic congestions are considerable. The traffic index which represents the ratio of real travel time to optimum travel time in 2017 was 1.899 and led to 4013.31 RMB congestion cost per person per years. The congestion index even raised to 2.165 in 2019 [4].

Traditional taxi plays an important part of urban transportation, which contains two types services Cruise taxi and Point taxi. However, its shortages, such as low efficiency and unstable price, constrain its development to satisfy the increasing demands of transportation, while the appearance of the online ride-hailing services, e.g., Didi Chuxing, changes this situation. The online ride-hailing service uses the GPS location system in smartphone to collect the location data of vehicles and users. Then using the dynamic dispatching algorithms, the online ride-hailing service alleviates the shortage problem.

Until the first quarter of 2019, Didi Chuxing’s has become the largest ride-hailing platform in China, whose users have reached 550 million and its daily route planning requests are more than 40 billion [5]. Didi Chuxing takes the full advantage of data mining and user matching technique, making users experience the efficiency of a modern transportation mode. Although a lot of ride-hailing apps existed when the online ride-hailing services appeared, Didi Chuxing survives with a market share of 88.4% and plays a dominant role in the market.

Although Didi Chuxing has such a huge market share, there are still some problems. On one hand, limited vehicle quantity still cannot satisfy the rush hour demands. On the other hand, drivers often drive with empty cars in the city. Therefore, demand prediction in different areas is needed. From the perspective of the company, the accurate prediction can help to dispatch vehicles flexibly, increase the utilization rate of vehicles, reduce the response time and increase user’s satisfaction. For drivers, an efficient dispatch regulation can prevent traffic jam, reduce the useless gas-consuming and increase the personal incomes. These can finally improve the social welfare and traffic efficiency.
To improve the performance of vehicle allocation, researchers have created many models based on
time sequence, machine learning and deep learning. The details of previous studies are discussed in the
related work section. However, this problem cannot be solved completely, there are two main reasons
for that. To begin with, many studies are based on traditional transport model, such as taxi, but not real
online ride-hailing data. This lead to the difference between research result and real situation. The other
reason is that the complexity of urban transportation differs in different cities, it is difficult to transfer
one city’s history experience to another city.

To address the two shortcomings mentioned above, we use the real data of Didi Chuxing in Haikou,
to discover the region-level ride-hailing of this coastal city. We propose a grid-based problem
formulation and verify the performance of several deep learning models in real dataset. The formulation
can be applied to other coastal cities.

In this paper, we focus on one important task: region-level ride-hailing demand prediction, which is
an essential part of the intelligent transportation system. Complex spatial and temporal correlations
contribute to the complexity of this task. On one hand, the complicate relationship exists between
different regions. For example, a CBD area’s ride-hailing demands are not only related to adjacent
regions, but also related to another CBD far away from it. On the other hand, the ride-hailing demand
in certain period has a relationship with others time segments, such as one day ago, one week ago, even
one month ago.

This paper is organized as follows. Section 2 introduces the related work. Section 3 describes the
formulation of the prediction problem. Section 4 introduces the detail about the dataset. Section 5
introduces the main models used in this paper. Section 6 concludes the experience process and major
findings. Section 7 gives the conclusion and future research directions.

2. Related Work

At the beginning, many researchers try to use some integrated framework which contains the features
selection to choose the most relevant variables and machine learning methodology to discover the hidden
regulation behind the demands. With the rise of deep learning techniques, they have become very
successful in prediction problems [6, 7]. There are generally two approaches, machine learning and deep
learning. For the machine learning approach, the features are usually extracted manually from the
experience. While for the deep learning approach, an end-to-end way is used and the features are
extracted automatically by the deep neural networks.

In this section, we introduce the related work from these two approaches.

2.1 Machine Learning Approach

In [8], the authors analyzed the sequence and travel data extracted from Didi Chuxing to quantitatively
determine the short-term spatial, i.e. at district level, temporal and so on. The authors use different types
of methods to obtain the prediction, such as Decision Tree (DT), Bagged Decision Tree (BDT), Random
Forest (RF), Artificial Neural Network (ANN) and Gradient Boosting Decision Tree (GBDT). With
experiments, they found GBDT showed the highest flexibility and efficiency in terms of data training,
with the lowest root mean squared error (RMSE).

In [9], the authors focus on the dataset from the on-demand mobility platform of DiDi Chuxing
between November 1 and 30, 2015, in Hangzhou, China. RF, Autoregressive Integrated Moving Average
model (ARIMA) and Support Vector Regression (SVR) are used to build the prediction models, in which
RF displays the best performance.

In [10], the authors analyzed the sequence and travel data extracted from Didi Chuxing. The
relationship between the different service modes of the driver and the selected area in a specific time
period is studied. The results show that RF performs better than other models and can predict the demand
for a unique on-demand ride-hailing service model.

In [11], the authors analyze the dataset of trip-level ride sourcing data in Chicago, tried to model and
predict demand for ridesourcing services in a large US city. In order to predict ride demand, the authors
use a direct demand model, classic multiplicative form and the random forest algorithm. After
comparison, they found random forest model is superior in terms of predictive accuracy and model fit.

2.2 Deep Learning Approach

Some common deep learning models, i.e., CNN [12] and LSTM [13], are already used for frames prediction. ConvLSTM [14] gives predictions in complex dynamical systems like the precipitation nowcasting problem. In moving-MNIST dataset, the authors input 10 frames to get the next 10 frame predictions. Compared with LSTM, it is able to build a network model not only for the precipitation nowcasting problem, but also for more general spatiotemporal sequence forecasting problems, and the predicted pictures are much clear than LSTM.

Predictive recurrent neural network (PredRNN) [15] makes the spatiotemporal memory interact with the original long short-term memory. This architecture is composed with Spatiotemporal LSTM (ST-LSTM) unit. It learns from representations extracted at different-level convolutional layers. In the Radar echo dataset, PredRNN has a better performance on MSE than ConvLSTM and VPN, and the training time is 12s higher than that of ConvLSTM (105s).

Improved predictive recurrent neural network (PredRNN++) [16] is used to improve the efficiency for deepen the recurrence depth and solve the vanishing gradient problem. In this model, it is constructed by causal LSTM and GRU, which process the spatial and temporal information separately for first. Compared with PredRNN, this approach has two advantages. First, it can derive a more powerful modeling capability to stronger spatial correlations and short-term dynamics. Second, it attempts to solve gradient back-propagation issues for the sake of long-term video modeling. In moving-MNIST dataset, PredRNN++ has the best performance. Its MSE is 20% lower than PredRNN.

In [17], the authors proposed Memory in Memory (MIM) blocks to learn about the higher-order non-stationarity from RNNs memory transitions. MIM shows the best performance in all predicted frames among all compared models, such as PredRNN and Causal LSTM, with the lowest difference intensities.

Eidetic 3D LSTM [18] is construct by two 3D-CNN and one E3D-LSTM layer in the middle, which can perceive and memorize both short-term and long-term representations in videos better. In the traffic flow prediction, author use E3D-LSTM model as the main method and the performance is much better than other methods, such as PredRNN and PredRNN++, with the lowest difference intensities in most areas.

GAN-LSTM [19] is for the cloud image training to generate diverse images with a low noise. The authors utilize LSTM to extract the implicit feature from the cloud image, and use the trained GAN to generate the predicted image. Compared with autoencoder-LSTM, GAN-LSTM model can efficiently decrease the dimension of the output data and improve the quality of the desired images.

We summarize the reviewed models and the compared baselines in Table 1.

| Article | Baselines | Proposed Mthod |
|---------|-----------|----------------|
| [8]     | DT, BDT, RF, ANN | GBDT |
| [9]     | ARIMA, SVR | RF |
| [10]    | ARIMA, SVR | RF |
| [11]    | Direct demand model, classic multiplicative form | RF |
| [12]    | Multi-layer perceptron, no-action feedforward | CNN |
| [14]    | FC-LSTM | ConvLSTM |
| [15]    | ConvLSTM, VPN | PredRNN |
| [16]    | PredRNN, FC-LSTM, ConvLSTM, TRAJGRU | PredRNN++ |
| [17]    | ST-ResNet, VPV, FRNN, PredRNN, Causal LSTM | MIM |
| [18]    | ConvLSTM, DFN, CDNA, FRNN, PredRNN, PredRNN++ | E3D-LSTM |
| [19]    | Autoencoder-LSTM | GAN-LSTM |
3. Problem Formulation
In this paper, we use a grid-based formulation approach for the region-level ride-hailing demand prediction problem.

In the literature, there are several types of problem definitions for online ride-hailing demand prediction problems, e.g., city-level, region-level, and spot-level predictions. For the city-level prediction, the focus is the demand in the whole city. This usually reflects the travel pattern without precise information at different geographic locations. The region-level prediction helps to schedule the travel demand and idle drivers at different regions within a city. The spot-level prediction only cares about the prediction at some hotspots, e.g., train stations and airports, which usually have a higher travel demand. Among them, we choose the region-level prediction, which is more related to the online ride-hailing service provided by Didi Chuxing.

In this paper, we use grids to divide spatial area into M*N part, and divide the time interval into T time steps. Then the ride demand in an area can be represent by $D(i, j, k)$ which represents the number of orders in grid $(i, j)$ in time slot k. This region-level ride-hailing prediction uses T historical demand logs before time slot k to predict all $D(i, j, k + 1)$ in the map, i.e.,

$$D(i, j, k + 1) = \arg \max p(D(i, j, k + 1)|D(i, j, k), D(i, j, k - 1), \ldots, D(i, j, k - T + 1))$$

4. Dataset
In this study, we use the online ride-hailing orders provided by the on-demand ride-hailing service company, i.e., DiDi Chuxing. The data is collected in Haikou, Hainan Province, China, covering a time range from May 1, 2017 to October 31, 2017. The spatial region of the dataset ranges in longitude from 110.204 to 110.396, and its range in latitude from 19.905 to 20.085. In this paper, we choose a prediction interval of 1 hour, and we use a 20*20 grid to divide the region into blocks. Each block corresponding to an area of 1 square kilometer.

In the dataset, the orders of Didi Express service account for more than 99% of the whole dataset, so we only choose Didi Express service as the experiment orders. Each transportation order contains the following fields: order id, starting time, origin longitude, origin latitude, destination longitude, and destination latitude. We do not use other fields of the original dataset. For all, we use 11,012,619 orders.

We show the order numbers at different timeslots on October 30, 2017 and October 31, 2017 in Figure 1 and Figure 2, respectively. From these two figures, we can tell that Didi’s service shows a daily pattern on the two consecutive days. Besides, the order numbers are higher at daytime than in the nights.
5. Models

5.1 Historical model
The history model has two types of prediction. The first type uses the previous one hour for prediction, which comes from the intuition that the demand may not change suddenly in two consecutive hours. The second one uses the same hour of yesterday as the prediction, which is inspired from the observation that the order numbers show a daily pattern. We denote these two historical model as history model1 and history model2 in Table 2, respectively.

5.2 Multi-layer Perceptron
Multi-layer perceptron (MLP) model is a typical structure of deep neural network, which has at least one hidden layer and uses activation function to achieve the non-linear feature learning ability. And it is a fully connected network. In this research, we use a MLP model with four layers, and each layer contains 512 neurons. We choose sigmoid as the activation function. Because the input of MLP model must be one-dimensional, we flatten each frame of data in one timeslot into the one-dimensional vector, as input for the MLP model. The output is also a vector. In this process, we neglect the spatial connection, which may loss the spatial information.

5.3 CNN
Convolutional neural network (CNN) is a kind of efficient neural network, which requires input be two-dimensional. Through operations including convolution and pooling, CNN can extract effective hidden features from two-dimensional input, e.g., frames in a matrix format. CNN training can be accelerated by the parallel computing ability of Graphics Processing Units (GPU). Because our dataset is two dimension in the geographic space, it is convenient to use CNN for the grid-based region-level prediction. Specially, we use a CNN model that consists of four convolutional layers, in which the convolution kernel has a size of 3*3. Batch normalization layers are used between two convolutional layers.

5.4 ConvLSTM
ConvLSTM model combines the convolution computation and Long Short-Term Memory (LSTM). In traditional LSTM, the model contains input gate, forget gate, unit, output gate and hidden state. In ConvLSTM, the neural network can capture the temporal dependency and spatial dependency in the dataset. Different from LSTM, the connection of input and different gates replaced by convolution, and the translation between states also replaced by convolution. This model not only can build time sequence model, but also can describe hidden features in convolution neural network. In this study, we use a ConvLSTM model that consists of four ConvLSTM layers, in which the convolution kernel has a size of 3*3. Batch normalization layers are used between two ConvLSTM layers.

6. Simulation

6.1 Settings
The OS used is Windows 10 family edition. We use Python and its packages (Tensentflow 2.3.0, NumPy 1.18.5, and Matplotlib 3.1.0) to implement these deep learning models. A GPU is used to accelerate the model training.

For deep learning models, we use the historical data from the past 6, 12, 18 or 24 hours as input frames to predict the next frame. Each frame represents the demand matrix in each timeslot. For training the models, we use Adam as optimizer and set learning rate as 0.001. We also set the batch size as 10 and the deep learning model’s epochs are set as 100.

6.2 Evaluation Metrics
We used the last 30 days of the dataset as the test set and the rest of the dataset as the training set. And we choose root mean square error (RMSE) as the evaluation metric. A smaller number of RMSE means
a better performance.

6.3 Results

The performance of different models is showed in Table 2. From table 2, it is clear to see that when the input historical length equals to 24 hours, ConvLSTM has the best result with the smallest RMSE. The ConvLSTM model itself has a better performance than others model when they have the same type of input. From Table 2, we can see that every model has a better performance when the input historical length is 24 hours.

Table 2. Experimental results of different models on the test set.

| Model          | Input Historical Length |
|----------------|-------------------------|
|                | 6 hours     | 12 hours     | 18 hours     | 24 hours     |
| History model1 | 6.337       |              |              |              |
| History model2 | 6.414       |              |              |              |
| MLP            | 5.808       | 5.278        | 5.272        | 5.267        |
| CNN            | 7.272       | 6.795        | 6.253        | 5.389        |
| ConvLSTM       | 5.162       | 4.812        | 5.148        | 4.796        |

7. Conclusion

In this paper, we propose different methodologies, such as the history model, MLP, CNN and ConvLSTM, to forecast the region-level ride-hailing demands. Methodological results have been discussed in details by using RMSE. The ConvLSTM model shows the highest accuracy in term of data training. When the input historical length is 24 hours, ConvLSTM model has the lowest RMSE (4.796). And in every circumstance, ConvLSTM model’s performance is better than the benchmark models, showing a more promising application prospect.

In current studies, the research scope is constrained in one city and region-level prediction. In the future, we may apply these methods into others cities and goes into details of the cities. For example, we can enable the demand prediction to district level or some shopping mall. And we can try to develop other kind of deep learning models and collect more data. Alternatively, we can try the transfer learning approach to apply the experiences to other cities, so that we can save a lot time and energy for training new models.

References

[1] Nation Bureau of statistics annual data (in Chinese). Available: http://data.stats.gov.cn/easyquery.htm?cn=C01. Accessed: Aug 12, 2020.
[2] China Urban Research Report 2017 (in Chinese). Available: http://huiyan.baidu.com/cms/report/Q4_niandu/index.html. Accessed: Aug 12, 2020.
[3] China Urban Vitality Research Report 2017 (in Chinese). Available: http://mapopencdn.bcebos.com/cms/report/2019niandu/ Accessed: Aug 12, 2020.
[4] China Urban Transportation Report 2019 (in Chinese). Available: http://huiyan.baidu.com/cms/report/2019annualtrafficreport/index.html. Accessed: Aug 12, 2020.
[5] Didi Chuxing’s Quarter Report (in Chinese). Available: https://sts.didiglobal.com/static/reports/2019Q1%E6%BB%B4%E6%BB%B4%E5%87%BA%E8%A1%8C%E4%BA%A4%E9%80%9A%E6%8A%A5%E5%91%8A-Final.pdf. Accessed: Aug 11, 2020.
[6] Jiang W, Zhang L. Geospatial data to images: A deep-learning framework for traffic forecasting[J]. Tsinghua Science and Technology, 2018, 24(1): 52-64.
[7] Jiang W. Applications of deep learning in stock market prediction: recent progress[J]. arXiv preprint arXiv:2003.01859, 2020.
[8] Saadi I, Wong M, Farooq B, et al. An investigation into machine learning approaches for forecasting spatio-temporal demand in ride-hailing service[J]. arXiv preprint arXiv:1703.02433, 2017.

[9] Faghih S S, Safikhani A, Moghimi B, et al. Predicting Short-Term Uber Demand in New York City Using Spatiotemporal Modeling[J]. Journal of Computing in Civil Engineering, 2019, 33(3): 05019002.

[10] Liu J, Cui E, Hu H, et al. Short-term forecasting of emerging on-demand ride services[C]//2017 4th International Conference on Transportation Information and Safety (ICTIS). IEEE, 2017: 489-495.

[11] Yan X, Liu X, Zhao X. Using machine learning for direct demand modeling of ridesourcing services in Chicago[J]. Journal of Transport Geography, 2020, 83: 102661.

[12] Oh J, Guo X, Lee H, et al. Action-conditional video prediction using deep networks in atari games[C]//Advances in neural information processing systems. 2015: 2863-2871.

[13] Srivastava N, Mansimov E, Salakhudinov R. Unsupervised learning of video representations using lstms[C]//International conference on machine learning. 2015: 843-852.

[14] Xingjian S H I, Chen Z, Wang H, et al. Convolutional LSTM network: A machine learning approach for precipitation nowcasting[C]//Advances in neural information processing systems. 2015: 802-810.

[15] Wang Y, Long M, Wang J, et al. Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms[C]//Advances in Neural Information Processing Systems. 2017: 879-888.

[16] Wang Y, Gao Z, Long M, et al. Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning[J]. arXiv preprint arXiv:1804.06300, 2018.

[17] Wang Y, Zhang J, Zhu H, et al. Memory in memory: A predictive neural network for learning higher-order non-stationarity from spatiotemporal dynamics[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 9154-9162.

[18] Wang Y, Jiang L, Yang M H, et al. Eidetic 3d lstm: A model for video prediction and beyond[C]//International Conference on Learning Representations. 2018.

[19] Xu Z, Du J, Wang J, et al. Satellite Image Prediction Relying on GAN and LSTM Neural Networks[C]//ICC 2019-2019 IEEE International Conference on Communications (ICC). IEEE, 2019: 1-6.