Taxi Demand Prediction using an LSTM-based Deep Sequence Model and Points of Interest

Bahman Askari
L3S Research Center
Leibniz University Hanover
Hanover, Germany
baskari@l3s.de

Tai Le Quy
L3S Research Center
Leibniz University Hanover
Hanover, Germany
tai@l3s.de

Eirini Ntoutsi
L3S Research Center
Leibniz University Hanover
Hanover, Germany
ntoutsi@l3s.de

Abstract*—Nowadays, urban mobility plays an important role in modern cities for city planning, navigation, and other mobility services. Taxicabs are vital public services in large cities that are taken by passengers thousands of times every day. Reducing the number of vacant vehicles on the streets will help service providers to raise drivers' incomes, reduce energy consumption, optimize traffic efficiency, and control air pollution problems in large cities. Since drivers do not have enough information about the location of passengers and other taxis, most of them might drive to the same area. Due to the lack of passenger information, they often end up without picking up any passengers while there are highly demanded areas in their neighborhood. To address these issues, machine learning techniques can be applied to analyze mobility data acquired from the IoT sensors and help companies to organize the taxi fleet or minimize the wait-time for both passengers and drivers in the city. In this paper, an LSTM-based deep sequence learning model is applied to forecast taxi-demand in a particular urban area in a smart city. For this purpose, points of interest (POIs) in the city are extracted from Google Maps and integrated with the mobility data sources. Given a real-world dataset and two evaluation metrics, we observed that taxi-demand in each urban area can be influenced by external factors such as neighborhood locations and the POIs located in that area. The results show that the proposed method outperforms the vanilla LSTM model and has less average error than baseline methods in terms of the Mean Squared Error (MSE) and Symmetric Mean Absolute Percentage Error (SMAPE).

***Index Terms*—Urban Mobility Prediction, Machine Learning, Deep Learning, LSTM, POI**

I. Introduction

A smart city is an urban area that uses different types of IoT sensors to collect data and enhance the quality and performance of urban services. Advances in IoT sensor and wireless communications such as Global Positioning System (GPS), Global System for Mobile Communications (GSM), and Wi-Fi have provided a new way of communicating with running vehicles while collecting relevant information on their status and location. Most of the taxi vehicles are now equipped with these kinds of technologies, producing a huge amount of rich Spatio-temporal information [4]. One of the most fundamental questions about smart cities is how to build prediction models to make better decisions in the future. Accurate taxi demand prediction can help service providers to reallocate taxis to highly-demanded areas, reducing both the waiting time of passengers and the vacant time of taxis. Thereby, increased taxi utilization could raise drivers' incomes, reduce energy consumption, optimize traffic efficiency, and control air pollution problems in large cities as in [5], [12], [15]. Since drivers do not have enough information about the location of passengers and other taxis, most of them might drive to the same area while there are higher demand areas in their neighborhood. Passengers also prefer to quickly find a taxi whenever they are ready for pickup. To address these issues a regression technique is presented in this paper to forecast the number of pickups in a particular urban area in the next short-horizon time. For this purpose and regarding to the first law of Geography which says, "everything is related to other things but near things are more related than distant things", the effect of points of interest (POIs) and mobility data sources are considered in each area of Porto. Furthermore, the sequential nature of taxi data needs a mechanism to store and remember the relevant information in future. Therefor, in this paper we use Long Short-Term Memory (LSTM) which is one of the best deep sequence learning methods for processing time-series data. The rest is organized as follows. Section II focuses on the related works. Section III presents the POIs (Points Of Interest) data set and the LSTM-based prediction model. Section IV introduces the Porto data set, then investigates the correlation between POIs and the taxi data and describes the experimental results. Finally, the most important findings of the research are highlighted in Section V.

II. Related Work

Most of the studies in urban mobility prediction split into two categories. The first group focuses on building learning techniques to forecast the mobility patterns in the future. Existing solutions of this group can be roughly classified into model-based and deep learning-based methods. Model-based methods such as ARIMA [14], Gaussian mixture [6], and linear regression [16] are simple to understand and implement, but they are not able to capture the complex spatial and temporal dependencies of taxi demand. In contrast, deep learning models can solve these issues and offer more powerful expressiveness for taxi demand prediction which is essentially considered as a time series processing problem [15]. The second group attempts to investigate the effect of external factors on the actual mobility patterns in a smart city. For this reason, POIs, weather conditions, and events are considered as external factors to add more relevant information to the learning process in prediction models. Lately, many traditional and Deep Neural Network (DNN) learning methods have been successfully used for processing sequential data and prediction tasks in a smart city as in [1]–[4], [8], [10], [19]. Recurrent neural networks (RNNs), special types of DNN, are popular models that can process sequential data very well. The idea behind RNNs is to store relevant parts of the input and use this information while predicting the output in the future. Unlike feed-forward neural networks that predict the output only based on the current input, RNNs contain memory in which some important information from the past inputs can be stored. The LSTM is a type of RNNs and a widely used deep sequence model in many applications such as time series predictions which can handle the spatial information and consider time dependencies for future uses. Furthermore, training time and performance of DNN-based solutions are extremely better than Non-DNN-based techniques [2], [9], [11], [13]. Contrary to most of the existing works replying solely on taxi-stand's own demand pattern, we enrich the data of each stand with the information of POIs closed to that stand. Furthermore, While most of similar works have divided the city in some static areas [2], [3] our method uses directly the taxi-stands information based on the traffic configuration of Porto city.

III. Methodology

This section is divided into three parts. First, the points of interest are defined and extracted. Next, two-dimensional (2D) time series are represented with a mathematical notation, and finally the LSTM deep sequence model is presented and evaluated.

A. Points of Interests

POI stands for Point of Interest, a kind of external factors, which describes popular places (e.g. bars, restaurants, shopping centers, etc.) in a city that might be interesting locations for visitors. Based on the first law of Geography, people may take a taxi or use the public transportation system to visit nearby locations as their pick-up and drop-off points. In this work, the geographical information of Porto POIs is extracted from the Google Maps service in order to be used as an external factor for the mobility prediction problem. Google Maps stores over 100 million places and their information such as location, contacts, user ratings, popular times, addresses, phone numbers, etc. In order to find POIs around a specific location, the GPS Coordinates and a radius is given to the Google API and it returns the nearby places corresponded to the given radius. There is also a parameter called "types" which can filter out only the types of places that are needed to find. For each taxi stand, a dynamic zone is defined as a circular area on the map. The radius of each circular zone is the distance between the current stand and the closest taxi station while the center of the circle indicates a taxi station. Around ninety types of places such as bar, restaurant, hospital, coffee shop, and school are listed to be searched in each zone. By doing so, the POI dataset is created for popular places in Porto. Each record consists of a unique id, name, address, coordinates and popular time to represent a place. In addition to this information, the id of the central taxi stand is assigned to the POIs located in the same zone and consequently, these zones are reshaped by assigning new locations or POIs. However, some circular areas have overlap, meaning that the POI-DATASET has redundancy. For this purpose the POI-DATASET is cleaned by removing duplicate records. Algorithm 1 shows the procedure of extracting POI data from the Google Maps.

```
Algorithm 1: Finding POI in Porto
  Input: Taxi-Stands Data TS , Business-Types BT
  Output: POI-DATASET
  for each stand i in TS do
      Radius[i] ← min(distance(i, j for each stand j in
      TS and i!=j))
      Coordinates[i] ← coordinate(stand i)
  for each b in BT do
      for each t in TS do
          p ← search_places_by_coordinate
          (Coordinates[t], Radius[t], BT[b])
          POI ← populartimes.get_id(API-KEY, P)
      Add POI to POI-DATASET
  Clean POI-DATASET //remove duplicate records
```

Both Taxi and POI datasets have a sequential structure because they strongly depend on location and time order. In order to apply sequence learning models, these data must be transformed and converted from their original format to time series format. The best aggregation period is 30 minutes based on the average waiting time at each taxi-stand [4]. Therefore, the whole year is divided into 17520 timestamps that each value indicates the number of pickups at each timestamp in Ps . Respectively two parallel time series are generated from the taxi data and POI in which the length of each time series (L) is 17520 according to Algorithm 2.

```
Algorithm 2: Converting POI-Dataset to time series
  Input: POI-DATASET PD
  Output: POI Time series Ps(Len (TS), L)
  // TS: Taxi-Stands , L: Length=17520
  for each poi in PD do
      visitlist ← convert POI to time series
      s ← index of closest taxi stand to the POI
      aggregate (Ps(s,:), visitlist)
  return Ps
```

B. Mathematical Notation and Data Transformation

Let define $\mathrm{S} = \{\mathrm{s}_1, \mathrm{s}_2, \ldots, \mathrm{s}_N\}$ and $\mathrm{X_s} = \{\mathrm{X_{s,0}}, \mathrm{X_{s,1}}, \ldots \mathrm{X_{s,t}}\}$, where S is the set of taxi-stands and $\mathrm{X_s}$ is a discrete-time series that models the taxi-demand for stand s. When the aggregation period is 30 minutes, $\mathrm{X_{s,i}}$ represents the number of pick-ups in stand s at i^{th} timestamp. The goal is to build a deep sequence model that predicts the demand $\mathrm{X_{s,t+1}}$ for the next time point $t+1$ at taxi-stand s where $\mathrm{X_{s,t}}$ is available. Traditional learning approaches use the demand history of the stand $\mathrm{X_s}$ for the prediction in a taxi station. In addition to these historical data, POI data will be considered in this work to improve the training process in the prediction model. In the following, the procedure of creating the POI time series is described by aggregating all points of interest around each station. Like taxi data, let assume $POI_s = \{P_{s1}, P_{s2}, \ldots, P_{sM}\}$ and $\mathrm{P_{sm}} = \{\mathrm{p_{s,0}, p_{s,1}, \ldots, p_{s,t}}\}$, where $\mathrm{POI_s}$ is the set of M points of interest in the neighborhood of taxi stand s and P_{sm} is the corresponding time series for point of interest m which has been assigned to stand s. Then $p_{s,i}$ represents the number of people visiting P_{sm} at timestamp i and the dynamic zones are reshaped by assigning each POI to the closest station and aggregating their time series in the P_s (number of people who are visiting the $\mathrm{POI_s}$ around taxi station s). Finally the P_s and X_s are concatenated to create the 2D time series XP_s. Fig. 1 illustrates a sample of the above-mentioned time series. For example, the number of pickups at timestamp 17519 is 1, while 4 people are visiting the points of interest in the nearby taxi stand s.

C. POI-LSTM Deep Sequence Model

The POI-LSTM stands for points of interest with LSTM, an extension of the well-known Long Short-Term Memory (LSTM) [17], which is a special kind of a recurrent neural network (RNN). The LSTM unit includes a cell state and a gate mechanism such that the cell state remembers values over arbitrary time intervals and the three gates regulate the information flow for each cell. When the LSTM model takes the time series data, the cell architecture stores the data for a specific period to train the model with back-propagation through time. Basically, the LSTM-based models use the logistic activation functions during the training process. Intuitively, the input gate decides which value flows into the cell, the forget gate controls the extent to which value remains in the cell and the output gate controls the extent to which the value in the cell is used to compute the output activation of the LSTM unit. There are connections into and out of the LSTM gates that a few of them are recurrent. The weights of these connections, which need to be learned during training, determine how the gates operate. However, the POI-LSTM deep sequence model is trained for each taxi-stand s by using the 2D time series which mathematically notated by XP_s in the Algorithm 3. Since the time-series data are different for each taxi stand, the model is separately trained and tested for all taxi-stands, and the final results are calculated by averaging all output values.

IV. Experiments and Evaluation

We perform the experiments on the historical data acquired from the Porto city, Portugal which is described in Section IV-A. The experimental setup and evaluation metrics are presented in the Sections IV-B and IV-C respectively. Finally, the prediction model is tested on two versions of the Porto dataset in Section IV-D and the results are compared with baseline methods.

Algorithm 3: POI-LSTM Deep Sequence Model

Input: Taxi-Stands **TS, Xs, Ps**
Output: POI-LSTM Prediction Model
for *each stand s in TS* **do**
  XPs← $Concatenate(Xs, Ps)$
  Train_s, Test_s← $Split(XPs)$
  Build and train LSTM with Train_s
  Execute prediction model with Test_s
  Evaluate the POI-LSTM model for stand s
return average(the evaluation metrics of all stands)

Xs	2	4	3	5	0	0	1	...	7	8	3	2	6	0	**1**
Ps	8	9	6	9	4	0	1	...	8	6	7	5	9	2	**4**
ts	0	1	2	3	4	5	6	...							**17519**

Fig. 1. 2D time series XP_s for a specific station s

A. Data Description

In this work, the trajectory data of a taxi company operating in the city of Porto, a medium-sized urban area in Portugal, was used as the case study. Due to lack of information, taxi-drivers in Porto waste a lot of time and energy to pick up a customer, consequently there is a huge competition between both companies and drivers to improve their services and obviously enhance their income. The existing regulations force the drivers to choose a specific taxi stand out of the 63 existing stations in the city and wait for the next service immediately after the last drop-off. The Porto dataset includes 1,710,670 taxi trips that continuously collected from 442 taxis operating through a taxi dispatch central, using mobile data terminals installed in the vehicles [14]. Each trip is represented with eight attributes in which "CALL TYPE" may contain three possible values A, B, and C that respectively means dispatched from the central service, demanded directly on a specific stand, and demanded on a random street by handshaking. "ORIGIN STAND" contains a unique identifier for the taxi stand when the corresponding "CALL TYPE" is equal to B, otherwise, it assumes a NULL value. When the GPS data stream is complete, "MISSING DATA" is False, and whenever a technical problem is caused to miss one or more locations, the corresponding "MISSING DATA" is set by TRUE value. Each pair of coordinates in "POLYLINE" represents the longitude and latitude of a location over a trajectory path.

B. Experimental Setup

Considering the "CALL TYPE" attribute, two versions of the dataset are created for the experiments. D1 is the small dataset including all trips departing from a specific taxi-stand.

This dataset contains 817.861 records and can be used in the prediction model that can forecast the short-term demand for specific taxi-stands. Around fifty percent of taxi trips are taken on the streets or by central service, and their information cannot be easily omitted. Therefore, the D2 version contains all records of the clean data. In this case, for those trips that do not start from a taxi-stand (i.e., those with a "CALL TYPE" equal to 'A' or 'C') the distance between their pickup location and all 63 taxi-stands is calculated, and the corresponding "ORIGIN STAND" value is replaced by assigning the closest taxi-stand id. By doing so, we define 63 dynamic zones where taxi stations are the centers of these regions. Fig. 4 shows the distribution of D1 and D2 at each taxi stand. In order to find the optimal POI-LSTM model, different parameters are used as follows:

- Layers: from range (1, 4) with step 1
- Neurons: from range (10, 300) with step 10
- Look back value: from range (2, 24) with step 1 where each step is 30 minutes.
- Epoch and batch size: from a list of possible values in range (10, 15, 20, 25, 50, 100, 200, 500, 1000)
- Dropout: from range (0.1, 0.9) with step 0.1

After testing the aforementioned parameters, we observed that the optimal model has one hidden layer including 200 neurons per layer. The best look-back value is five as it raises the best value of SMAPE. The AdamMax and tanh are selected for the gradient descent optimization algorithm and activation function respectively because they cause the best SMAPE values compared to other functions. Additionally, the best values for epoch and batch size are set as 25 and 100 respectively. To prevent overfitting in POI- LSTM model, the dropout technique is set to 0.7 to randomly drops units and their connections from the neural network during the training part. Using different time intervals we showed that the POI-LSTM achieves the best performance when 30-minutes timestamp is set rather than for example 60 minutes. Therefore, 30 minutes is chosen as the horizon time for the prediction problem. We use the first 70% of each data set for training, the remaining one is used in the test phase.

C. Evaluation Metrics

Evaluation measures in time series prediction represent the capability of a model for the prediction task. Two typical metrics are used in this work to compare all methods: Mean Squared Error (MSE) and Symmetric Mean Absolute Percentage Error (SMAPE) which are given by equations (1) and (2). The SMAPE is more meaningful than the MSE because the proportion values are more comprehensive than squared errors. The constant value c in corrected equation (2) is a user-defined value which prevents high error just in case the real demand is 0 and the predicted one is non-zero.

$$\mathrm{MSE}_s = \frac{1}{T}\sum_{i=1}^{T}\left(Y_{s,i} - \hat{Y}_{s,i}\right)^2 \tag{1}$$

$$SMAPE_s = \frac{100\%}{T}\sum_{i=1}^{T}\frac{\left|Y_{s,i} - \hat{Y}_{s,i}\right|}{|Y_{s,i}| + \left|\hat{Y}_{s,i}\right| + c} \tag{2}$$

Where $Y_s = \{Y_{s,0}, Y_{s,1}, .., Y_{s,t}\}$ and $\hat{Y}_s = \left\{\hat{Y}_{s,0}, \hat{Y}_{s,1}, .., \hat{Y}_{s,t}\right\}$ are real and predicted demand values of stand s by the regression model. T indicates the length of time series which is 17520 in this case. The aforementioned formulas refer to the error rate at each taxi stand, and they need to be averaged over all taxi-stands as follows:

$$MSE = \frac{\sum_{i=1}^{N} MSE_i}{N} \tag{3}$$

$$SMAPE = \frac{\sum_{i=1}^{N} SMAPE_i}{N} \tag{4}$$

Where N is the number of taxi-stands.

D. Results and Discussion

To find points of interests in Porto, a list of ninety business types such as museum, school, hospital was explored. After extracting and removing redundancy, 12851 distinct places were remained, but the Google Maps does not store popular times (based on visits to a place) information for all places. Considering this restriction, only 2051 distinct points of interests have the popular time information and more than 10000 places are not usable to form the POI time series in this case. For instance, 432 bars were found in Porto but only 122 of them include popular time list which represent the number of people who visit this place during the day and weekdays. POIs are automatically extracted from the Google Maps and they may not reflect the exact number of visitors because it does not record everything, but interestingly the number of POI including popular times data and the number of places without this information have the same trends in each urban area. In an urban area, spatial and temporal features are two main mobility characteristics that reflect the geographic location and regularities in the time dimension respectively [14]. Concerning the spatial dimension, the correlation between the number of pickups and visitors has been shown in the Fig. 2. This graph visualizes the spatial distribution of the total number of visitors and pick-ups in each zone from 01/07/2013 to 30/06/2014. The blue color indicates the number of people who visit POIs while the red line is used to represent the number of pickups in a specific zone. The results show that, generally the number of visitors are more than the number of pickups in most of stations. On the other hand, in a some stations the difference between the number of visitors and pickups is not significant, meaning that the POIs information is not able to precisely reflect the correlation between visitors and using taxicabs by them. For instance Fig. 2 shows that taxi stands 7 and 42 have less number of visitors than the number of pickups during the whole year in Porto while stand 15, the central train station, has more than 100,000 pickups during the whole year which is almost equal to the number of visitors. One of the reasons for the irregular fluctuation in spatial distribution in Fig. 2 can be due to the lack of popular times in POIs data.

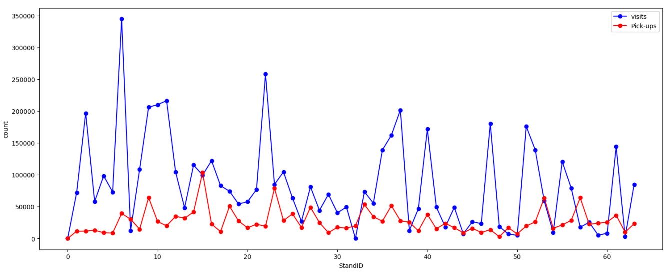

Fig. 2. Number of visitors and pickups in each area during the whole year

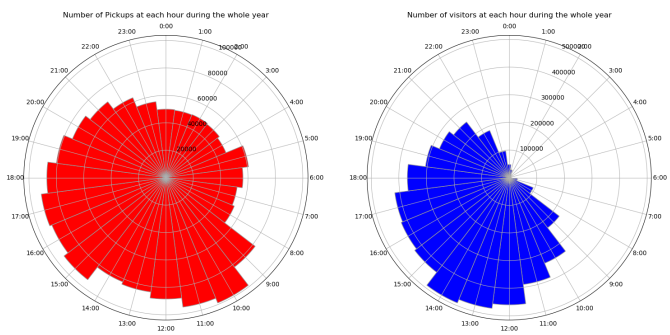

Fig. 3. Number of Pickups and POIs over 24 hours (temporal distribution)

The temporal distribution of both visits and pickups for POI and taxi data have been represented in Fig. 3. The red pie chart shows the temporal distribution of pickups and the blue indicates the same distribution for visitors. The whole day is divided into 24 time-interval from 0 to 23 in a clockwise direction around the pie chart. The number of visitors and pickups are aggregated during the whole year at each time-interval and their statistics are separately represented using five concentric circles in the pie charts. By doing so, the temporal correlation between both time series data can be shown like the spatial correlation in Fig. 2. It is seen from Fig. 2, the red pie chart, that the number of pickups has the same distribution from midnight to 4 am, then it is increased until 8 am. It can be guessed that people usually go out from home at around 9 am meaning that they may take a taxi for moving in the city. Consequently, the number of pickups has a marked deference at 9 am and reaches to a high peak at 10 am. After fluctuation briefly between 10 am and 7 pm, the number of pickups is decreased until midnight. Approximately the same trend can be observed for the POIs during the 24 time-interval of the days. All in all, spatial and temporal correlation between historical taxi data and POIs, and concatenating them as a 2D time series can be a meaningful input for the deep sequence learning model.

The network is trained using both D1 and D2 dataset with the same parameters described in the Section IV-B and the POI dataset. Three traditional models such as linear regression (LR), random forest (RF), and XGboost regression are used for comparing with the LSTM-based models. Both traditional models and vanilla LSTM were trained using the historical taxi data. Generally, the vanilla LSTM model outperforms traditional prediction models. The neighborhood-augmented LSTM (NA-LSTM) [18] is an LSTM-based deep sequence model that consider a global neighborhood threshold k for all taxi-stands to improve the performance of the LSTM model. Table I depicts the predictive performance of the traditional models and the LSTM-based models for data set D1, containing trips starting from an actual taxi-stand. In this table, the model is fitted with 30 minutes timestamp. Table II summarizes the results for data set D2, containing all trips from the cleaned data set and after mapping the trips that do not start from a stand to their closest stand. The MSE and SMAPE metrics in Tables I and II show that, the proposed POI-LSTM Model has the smallest error values for both D1 and D2 where again the prediction problem was fitted for 30 minutes timestamp. As indicated, the Random forest regression has less error in training process, but this error is significantly increased when the model is tested for unseen data. This clear difference between train and test performances meaning that the models overfits, instead of learning from data. POI-LSTM has the least error over unseen data and these results are much better when the D1 dataset is used for the experiments. A possible reason is the assignment of the trips to their closest taxi-stands in dataset D2. Fig. 5 shows the SMAPE error rates for training and testing processes over dataset D2. As an extreme case, the most popular stand, stand 15 corresponding to the main train station, has the highest error. A possible explanation is that such a stand is very difficult to model with a single model and one might need to consider different models for different contexts such as season based, weekdays vs weekends etc. In general, POI-LSTM model has higher improvement when it is run on D2 data set and experimental results demonstrate the superior performance of the proposed model as compared to the existing approaches.

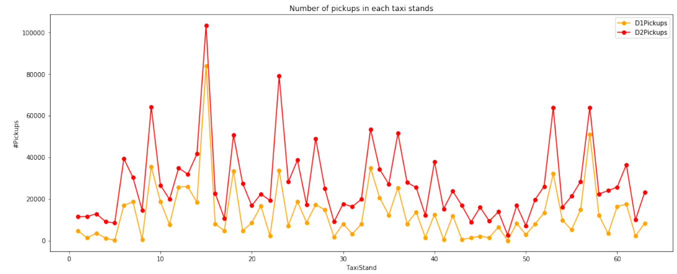

Fig. 4. Number of pickups in D1 and D2 at each stand (spatial distribution)

V. Conclusion and Future Works

In conclusion, an LSTM-based deep sequence model was used in this study to predict taxi demand in a smart city. It was found that the proposed model can provide more accurate results compared to the baseline models for the motioned purpose. The model was able to utilize the additional information of external factors to improve the learning process.

TABLE I
PERFORMANCE OF THE DIFFERENT MODELS ON D1

Model	TrainMSE	TestMSE	TrainSMAPE	TestSMAPE
LR	1.61	1.76	24.37	24.52
RF	0.38	1.66	16.83	24.25
XGBoost	1.39	1.59	23.90	23.91
LSTM	1.66	1.84	18.37	18.54
NA-LSTM	1.49	1.68	17.32	17.64
POI-STM	**1.26**	**1.41**	**17.12**	**17.25**

TABLE II
PERFORMANCE OF THE DIFFERENT MODELS ON D2

Model	TrainMSE	TestMSE	TrainSMAPE	TestSMAPE
LR	4.21	5.99	30.78	31.23
RF	0.71	5.50	18.49	31.03
XGBoost	3.60	5.45	30.47	30.51
LSTM	4.16	6.66	27.03	27.22
NA-LSTM	3.84	6.44	*25.88*	26.07
POI-STM	**2.57**	**5.27**	**24.73**	**25.08**

Given the historical taxi data, we observed that taxi-demand in each area can be influenced by neighborhood locations and the POIs located in that area. POI data set includes useful information for the peak hours, but for the off-peak hour we need more external information from other resources because the performance of prediction models are tightly connected to the input features. Therefore, considering more relevant features such as events and climate conditions for improving the performance of the model can be done as a future work. In this paper, the main focus was on investigating the effects of the external factors to improve the basic prediction model. Another solution can be improving the learning algorithm and building a more accurate prediction model. Decision making by inappropriately trained algorithms may unintentionally discriminate the results and the algorithms may capture and propagate ethnicity related biases. Therefore fundamental machine learning principles can be used to prevent discrimination issues for achieving fairness in taxi demand problem.

ACKNOWLEDGMENT

The contribution of Bahman Askari was carried out during his Erasmus internship at the L3S Research Center of LUH.

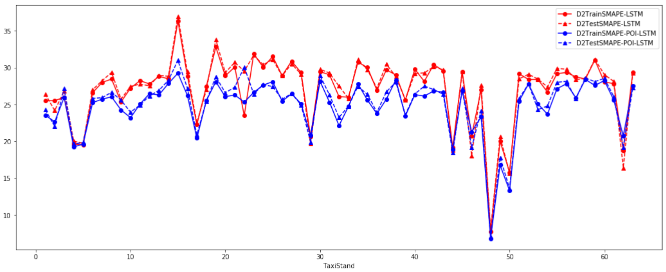

Fig. 5. SMAPE error rate over D2

REFERENCES

[1] J Yuan, Y Zheng, L Zhang, X Xie, and G Sun, “Where to find my next passenger,” UbiComp'11, Beijing, China, pp. 109–118, September 2011.

[2] J Xu , R Rahmatizadeh , L Bölöni, and D Turgut, “Real-time Prediction of Taxi Demand Using Recurrent Neural Networks,” IEEE Transactions on Intelligent Transportation Systems, vol. 19, pp. 2572 - 2581, August 2018.

[3] L Cagliero, S Chiusano, E Daraio, and P Garza, “CarPredictor: forecasting the number of free floating car sharing vehicles within restricted urban areas,” IEEE International Congress on Big Data (BigData-Congress), Milan, Italy, July 2019.

[4] L Moreira-Matias , J Gama , M Ferreira , J Mendes-Moreira , and L Damas, “Predicting taxi–passenger demand using streaming data,“ IEEE Transactions on Intelligent Transportation Systems, vol. 14, pp. 1393 - 1402, September 2013.

[5] M Catalano, F Galatioto, M Bell, A Namdeo, and A.S Bergantino, “Improving the prediction of air pollution peak episodes generated by urban transport networks,“ Elsevier, Environmental Science Policy, Vol. 60, pp. 69-83, June 2016.

[6] C Meng-Fen, H Tuan Anh, and L Ee-Peng, “Where are the passengers?: A grid-based Gaussian mixture model for taxi bookings, “SIGSPATIAL International Conference on Advances in Geographic,“ SIGSPATIAL '15: Proceedings of the 23rd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Seattle, Washington, November 2015.

[7] T.M Mitchell, Machine Learning, McGraw-Hill, 1997.

[8] L Moreira-Matias et al., “Data4UrbanMobility: towards holistic data analytics for mobility applications in urban regions,“ Companion Proceedings of The Web Conference, March 2019.

[9] P Li,M Sun, and M Pang, “Prediction of taxi demand based on ConvLSTM neural network,“ 25th International Conference, ICONIP 2018, Siem Reap, pp. 15-25, Cambodia, December 2018.

[10] Q Zou, G Xue, Y Luo, J Yu, and H Zhu, “A novel taxi dispatch system for smart city,” International Conference on Distributed, Ambient, and Pervasive Interactions, pp. 326-335, July 2013.

[11] S Liao, L Zhou , X Di, B Yuan , and J Xiong, “Large-scale short-term urban taxi demand forecasting using deep learning,“ IEEE 23rd Asia and South Pacific Design Automation Conference (ASP-DAC), Jeju, South Korea, January 2018.

[12] T Maniak, F Doctor, and R IqbalEmail, Advances in Hybridization of Intelligent Methods, pp. 57-70, 2017.

[13] U Vanichrujee, T Horanont, W Pattara-atikom, T Theeramunkong, and T Shinozaki, “Taxi demand prediction using ensemble model based on RNNs and XGBOOST,“ IEEE International Conference on Embedded Systems and Intelligent Technology International Conference on Information and Communication Technology for Embedded Systems (ICESIT-ICICTES), Khon Kaen, Thailand, May 2018.

[14] X Li et al., “Prediction of urban human mobility using large-scale taxi traces and its applications,“ Frontiers of Computer Science, Vol. 6, pp. 111–121, January 2012.

[15] Y Xu and D Li, “Incorporating graph attention and recurrent architectures for city-wide taxi demand prediction,” International Journal of Geo-Information, Vol. 8, September 2019.

[16] Y Tong et al., “The simpler the better: A unified approach to predicting original taxi demands based on large-scale online platforms,“ KDD '17: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1653–1662, New York, United States, August 2017.

[17] S Hochreiter and J.U Schmidhuber, “LONG SHORT-TERM MEMORY,“ Neural Computation, Vol. 9(8), pp. 1735-1780, 1997.

[18] T Quy, W Nejdl, M Spiliopoulou, and E Ntoutsi, “A neighborhood-augmented LSTM model for taxi-passenger demand prediction,“ MASTER2019 Workshop@ECML PKDD, Würzburg, Germany, 2019.

[19] J Chu et al., “Passenger demand prediction with cellular footprints,” 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), Hong Kong, China, June 2018.