A Hybrid Convolutional Approach for Parking Availability Prediction

Hadi Samer Jomaa
Information Science and Machine Learning Lab
University of Hildesheim
Hildesheim, Germany
hsjomaa@ismll.uni-hildesheim.de

Josif Grabocka
Information Science and Machine Learning Lab
University of Hildesheim
Hildesheim, Germany
josif@ismll.uni-hildesheim.de

Lars Schmidt-Thieme
Information Science and Machine Learning Lab
University of Hildesheim
Hildesheim, Germany
schmidt-thieme@ismll.uni-hildesheim.de

***Abstract*—Parking availability prediction is rapidly gaining interest within the community as an operationally cheap approach to identifying empty parking locations. Parking locations accommodate multiple vehicles and are rarely completely occupied. This makes it difficult to predict occupied locations without the augmentation of external data, as the data becomes highly imbalanced. Existing forecasting models neither encapsulate the heterogeneous modes/types of parking data, nor can handle sparse measurements. The problem is formulated as a binary forecasting task, based on the parking occupancy information. In this paper, we propose a new convolutional hybrid model that is capable of capturing long term temporal dependencies and outperforming conventional time-series forecasting benchmarks on two types of parking data, namely on- and off-street parking. The performance of the proposed model is further boosted by integrating external features such as location identifiers, as well as local/global statistics. An extensive experimental evaluation proves that the proposed model is capable of handling sparse data by maintaining high precision and recall across different sparsity levels, which are controlled by empirically adjusting the occupancy cut-off threshold, as well as for multiple horizons, with an average F1 score improvement of 4.13% over strong off-the-shelf baselines.**

***Index Terms*—Data science, parking availabiltiy, time series forecasting, Recurrent neural networks, convolutional neural networks**

I. Introduction

The availabilty of a parking location plays a major role in private car based trip decisions, and is governed by multiple factors including the time of day [1], the location of the target destination [2], traffic [3], weather [4], local events [5], safety, etc. This can be a demanding task in big cities, specifically in those where drivers are not accustomed to. Finding a place to park only increases in difficulty with time, as studies in 2013 suggested that around 25 billion car trips occur annually, with every car spending on average 162 hours a week being parked [6]. Searching for the 'right' spot, epecially in urban areas, leads to higher rates of congestion, as drivers tend to reduce their speed [7] as they approach an on-street parking spot, or when cruising the block to explore their options. More than one-third of traffic in big cities is caused by searching for a parking space [8]. Circling around also harms the environment, with emissions leading to higher rates of air pollution, not to mention the extra expenses accumulated on the driver, all the while leading to increased stress levels, in the search for the least possible walking distance.

Facilitating the parking process has led to the development of several parking guidance systems, that help the user identify available parking spots, along with other functionalities such as e-payment [9]. Most of the existing solutions depend on the continous collection of sensor data, in an Internet-of-things framework, where sensors are spread around metroplotian cities to indicate the occupancy of the respective slots. However, due to the expensive operational cost (US $2500 [10] to setup a single slot), such solutions [11], [12], lack scalability.

The availabilty of public parking data encourages the search for cheaper solutions that model parking occupancy as a regression problem, that aims to predict the current occupancy rate of a specified parking location. With the significant representational power of neural networks, feed-forward neural networks show impressive performance [13]–[15]. The inherent seasonality present in data is also leverage with autoregressive models, that can be based on multivariate [16] or univariate [17] input.

Existing state-of-the-art methods focus on recurrent neural networks for dynamic modelling, as well as heterogenous data for improved parking occupancy prediction. They also tackle the problem as a regression task, which in terms of parking recommendation, is not as critical as the ability to predict the status. Such methods require more computational power, leading to slower training, and do not handle datasets with significant imbalance well. We reforumulate the parking avail-

We would like to thank Dr. Alexander Borek (alexander.borek@vwfs.io) from Volkswagen Financial Service AG for his insightful input during our discussions, as well as for providing us with the necessary data from PayByPhone!

ability problem as a binary forecasting task, instead of a simple regression task. The assumption behind it is that drivers, in general, are not particularly interested in finding the percentage of availability in a given location; they are only interested in knowing whether or not they will be able to find an available spot for their automobile. It is important to note the distinction between a parking location and a parking spot. The former represents a geographic area, which can be on-street or off-street parking, that accommodate multiple cars, with each car occupying exactly one spot. The rate of occupancy is hence calculated based on the parking lots occupied, with respect to the total number parking lots at a given location. Attempting to model the dynamics of multiple parking locations can be tricky, as different locations can display different trends whereas training one model per location leads to overfitting and the averaged evaluation can mask the true location-based performance of the model.

In this paper, we propose a hybrid convolutional model that is able to predict the status, i.e. occupied or free, for multiple locations. The status of the location is set by an empirical threshold that mirrors our confidence in the groundtruth data. As far as we know, we are the first to propose such a solution, that captures the low-level temporal dynamics of the system using convolutional layers, and takes advantage of the location, as well as relevant time information, i.e. year, day, month, etc. The proposed model also exploits seasonality of the parking locations, by augmenting a set of statistics for every time step. Our model is tested on BANES[1] public dataset, as well as a private dataset provided by PayByPhone[2]. The results, as compared to LSTM's and Gradient Boosted Decision Trees highlight the performance of our models, and their ability to scale over long horizonsm, with an average F1 score improvement, across 36 distinct design paramaters, of 1.31% and 6.94%, respectively.

II. Related Work

Existing smart parking solutions are diverse [18], and can depend on user-provided data, for example: Parkopedia[3] or ParkNav[4], or parking data acquired from sensors, however, centralized system-based approaches fall outside the scope of this paper. We focus here on parking availability prediction, which has also received significant attention recently. Other models have also been proposed, which tackle availability modeling as a Markov chain [19]–[21], while others use clustering techniques [22]. Solutions that are based on artificial neural networks are also investigated. In [17], parking occupancy is first evaluated by grouping multiple parking spots together to form a parking location, and trained on feed forward network with two hidden layers. In [13], the proposed solutions includes an MLP with eight hidden layers on five-minute windows trained to predict 30 minutes into the future (each timestep represents one minute). In [15], they use an input of 1383 dimensions, that include binary features representing events that intuitively influence parking behavior as well as meteorogical information such as the temperature and rainfall.

Perhaps the closest approach to our model is proposed in [23]. In their work, parking availabilty is modeled as an integration of three major components, closeness, periodicity and general influence. The dynamics of the historical data are captured by two LSTMs that are concatenated with engineered features, i.e. weather, holidays, event, in an feed-forward neural network. The dataset and proposed feature used by [23] are not publicized, which prevents a fair comparison against that method. The status of the parking location is set empirically as well, depending on the point of interest.

In [24], the spatial aspect of temporal data is leveraged in a deep neural network based model. The architecture is composed of a spatio-temporal component, that captures the dynamics of the data via convolutional layers, and a global component that takes in external features, before fusing their respective outputs for regression. Another spatio-temporal approach is suggested by [25] for traffic prediction. The approach also capture spatial dependencies of the streets using a convolutional layer, while modeling the temporal dependencies using LSTMs. A diffusion convolutional recurrent network is proposed in [26], which is essentially consists of an encoder-decoder setup coupled with a sequence-to-sequence learning framework.

In contrast to existing models that require features collected from different sources, we propose a model that relies only on the univariate parking occupancy distribution. The convolutional approach is also better suited for sparser data, scales better for longer horizons, and requires much less training time.

III. Model

Parking availablity prediction is formulated in this paper as a time series binary forecasting problem. Given the time window of occupancy rates, X_t^i at a specific parking location, $i \in \mathcal{P}$ where $\mathcal{P}$ represents the total number or locations, and time t, the objective is to predict at horizon h, the status of that location, i.e. occupied or free. The status is set empirically by an occupancy-threshold, which we investigate in the Experiments Section. The problem is therefore formulated as

$$y_{t+h}^i = \Omega(X_t^i|\theta) \quad (1)$$

where i corresponds to a specific parking location, and θ represents the parameters of model Ω.

For our model, we use multiple one-dimensional convolutional layers to extract low-level representation from the time-series. For simplicity, we drop the location index as the method is trained on all parking locations uniformly. Let $X_t = \{x_1, x_2, \cdots, x_t\}$, with $x_i \in [0, 100]$ be a univariate sequence that represents the percentage of the total places in location being occupied, $X_t \in \mathcal{R}^{1\times t}$ and y_{t+h} the corresponding scalar representing the availability status at horizon h for any parking location $i \in \mathcal{P}$. Given a $1 \times m$ filter with weights w and the

[1]https://github.com/BathHacked/documentation/wiki/Bath-Car-Park-Data
[2]https://www.paybyphone.co.uk/
[3]https://www.parkopedia.com/
[4]http://parknav.com/

input to a convolutional layer l being x_i^l, we can compute the output of the convolutional layer as

$$x_{i,j,k}^{(l+1)} = \sigma \left(\sum_{r=0}^{m-1} w_{r,k} x_{i,j+r,k}^{(l)} \right) \tag{2}$$

with σ as a non-linear function, and k refers to the feature map.
Convolutional layers are capable of capturing the dynamic changes in the historical data. At the final layer, we apply global max-pooling across all the feature maps in order to keep the most prominent values. This representation, Q, is then introduced into a fully connected layer, with a sigmoid activation function, to predict the output probability. The model is split into two parts, the convolutional layer, which extracts temporal features from the time windows, and the fully connected layer that generates the output,

$$y_{t+h} = \Omega(Q|\theta) \tag{3}$$

$$Q_i = \frac{1}{J} \sum_{j=1}^{J} x_{i,j,k}^{L} \tag{4}$$

1) Location: Motivated by the first Law of Geography: "near things are more related than distant things" [27], and the fact that parking locations follow distinct parking patterns, we introduce the location identifier, L_i, as a one-hot encoded vector, to the latent representation obtained from the convolutional layers. This is done by appending the convolutional layer output with the location identifier, forming an extended representation. This enables the model to identify location-specific transitions, making the prediction y_{t+h}^i conditioned on the location as well as the historical data,

$$y_{t+h}^i = \Omega([Q_i, L_i]|\theta) \tag{5}$$

2) Statistics: Another aspect to parking is that it contains an inherent seasonality. People tend to park in the most convenient places, specifically those that they are accustomed to. To capture this seasonality, we compute the statistics of bins; each bin consists of five minutes and represents the smallest time unit in the processed series, taking the mean, maximum and variance across: (a) bins, (b) [bin,hour] pairs, (c) [bin,weekday] pairs, and (d) [bin,hour,weekday] tuples, up until the previous day of the pair/tuple in question, as presented in Figure 1. Statistics also include one-hot encoded time information. These statistics can be utilized in two ways, either by being a part of the input time-series, i.e. changing $X^i \in \mathcal{R}^{1 \times t}$ from a univariate series to a multivariate series, $X^i \in \mathcal{R}^{1 \times t \times s}$ where s is the number of statistical values, or by adding the statistics of the target point as part of the input, much similar to what is done at the location level, Figure 2, leading to a prediction multi-model feature fusion model of the form,

$$y_{t+h}^i = \Omega([Q_i, L_i, S_i]|\theta) \tag{6}$$

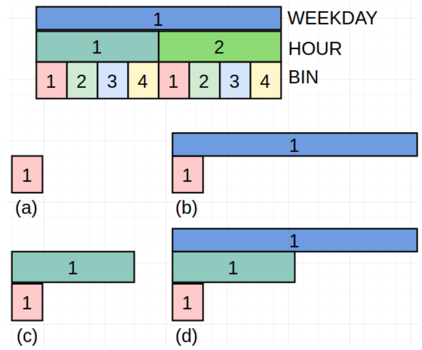


Fig. 1: Statistics

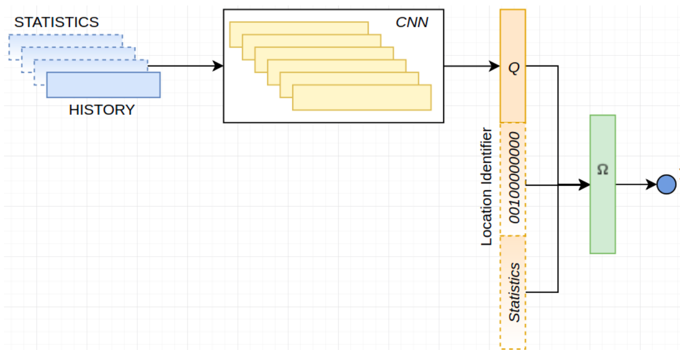


Fig. 2: Time series modeling with historical data; The features with disconnected borders are introduced as an extension to the base model.

IV. Model Architecture

The time series are trained in a convolutional block with four one-dimensional convolutional layers with a feature map size of 64 , 1×3 kernels, and ReLU activation functions. We apply a global max-pooling layer on the final feature map. The corresponding representation is then followed by two fully connected layers of size 512 and 256 respectively.
The extended models, which include location and/or statistical information, are set-up by simply appending the additional information to the output of the convolutional block, as shown in Figures 2.
As this is a binary forecasting problem, we optimize the model by minimizing the binary cross-entropy loss, prediction multi-model feature fusion model of the form,

$$CE = -\sum_{i}^{P} \sum_{t}^{T-h} y_{i,t+h} \log(\hat{y}_{i,t+h}) + (1 - y_{i,t+h}) \log(1 - \hat{y}_{i,t+h}) \tag{7}$$

where $y_{i,t+h}$ represents the ground-truth and $\hat{y}_{i,t+h}$ represents the predicted status at a location i at horizon h.

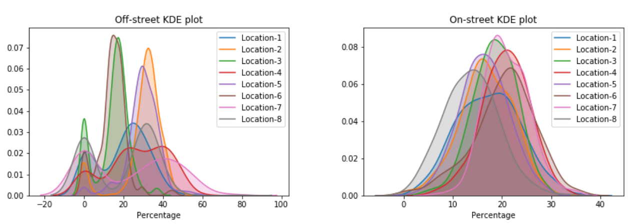
V. Experimental Evaluation

In this section we describe how the performance of the proposed model is evaluated for binary time-series forecasting.

A. Datasets

1) BANES Historic Car Park Occupancy: The publicly available BANES Historic Car Park Occupancy dataset includes occupancy rates of 8 different off-street parking houses in London, which are located next to each other. The data is updated in incremental steps of five minutes, a more meaningful resolution than one minute updates, in parking-related problems. We refer to these five-minute updates as bins throughout the rest of the paper. Figure 3 shows the kernel density distribution, which represents the probability distribution generated by the data, for both off- and on-street parking.

2) PayByPhone Parking Data: PayByPhone, PbP, is a parking assistant application that facilitates payment transactions at specific parking locations. As a user/driver, you specify the parking lot as well as the expected duration on a minute-based resolution before parking. This transaction is then registered and saved in the database. Each parking lot has a unique ID and a street ID, that is shared across neighboring parking lots. To unify the data representation with the off-street parking data, we first expand the transactions over the entire day before grouping the parking lots together based on the street ID. We then calculate the respective occupancy of every street. Due to the fact that some cars leave the designated spot before the termination of the pre-allocated time, transcations can overlap and lead to an occupancy of more than 100%. Setting a maximum of 100% occupancy rate is also part of the preprocessing. Finally, the data is aggregated over five-minute bins. The data supplied by PayByPhone also includes eight on-street locations with more than 1 million transactions that span more than two years in Vancouver, Canada.

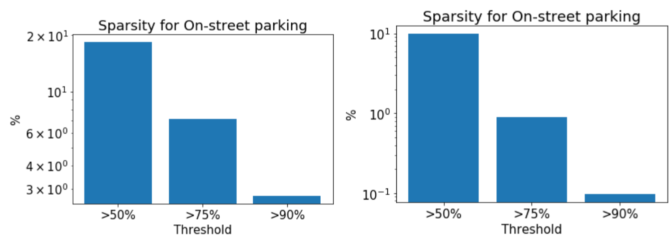


(a) Off-street Parking (b) On-street Parking

Fig. 3: Gaussian Kernel Density Estimate

B. Protocol

We trained our models using TensorFlow on the datatsets described above, independently, using a batch size of 64, a learning rate of .0001, and an AdamOptimizer [28]. We trained the models on two GeForce 1080 TI GPUs, and training ended when no improvement was detected on the validation set for 20 iterations. The same hyperparameters were used to train all models. For comparison, we implemented gradient boosted decision trees using the XGBOOST [29] library, and an LSTM with 128 neurons. The metrics used to evaluate the models are precision, recall and F1-score. Throughout the experiments, we investigate the performance over 6 different horizons: 5 minutes (1 time step ahead), 15 minutes (3 time steps ahead), 30 minutes (6 time steps ahead), 1 hour (12 time steps ahead), 2 hours (24 time steps ahead), and 3 hours (36 time steps ahead), as well as two history lengths, specifically: 4 and 8 hours. We also ran the same experiments with three different occupancy-thresholds. The threshold is set empirically to control the value at which we consider the parking location to be fully occupied. This threshold is set as a, 50%, 75% and 90% precautionary measure, that takes into consideration external factors that are not foreseen in the data itself, which might include illegal parking, non-digital monetary transactions, etc. The higher the threshold, the sparser the occupancy distribution.
As mentioned earlier, this results in an imbalanced dataset, which is highlighted in the bar graphs of Figure 4.



(a) Off-street Parking (b) On-street Parking

Fig. 4: Sparsity for parking.

C. Baselines

We compare our proposed model with two strong baselines. The comparative figures presented later in this section are between the baselines and the simple convolutional model proposed, where all three models take as an input a univariate occupancy window.

1) Gradient-boosted decision trees: Gradient-boosted decision trees (GBDT) is a widely adopted algorithm within the field of time-series forecasting. The goal of this approach is to choose a classification function $F(x)$ to minimize the sum of a specified loss $L(y_i, F(x_i))$, where prediction multi-model feature fusion model of the form,

$$F_{optimal} = \min_F \sum_{i=1}^{N} L(y_i, F(x_i)) \tag{8}$$

. This approach results in a prediction model in the form of an ensemble of weak prediction models. The algorithm is provided as an open source python library, that demonstrates fast and efficient learning.

2) Recurrent Neural Networks: Recurrent neural networks are an extenstion of deep neural networks, that is capable of handling data of sequential nature. We use long short term memory cells, LSTM, a variant of RNNs that are better suited to capture the temporal dependencies in time-series, where short- and long-term temporal dependencies exist. An LSTM consists of a forget gate layer f, an input gate layer i, and an ouput gate layer, o. Unlike feed-forward neural networks, LSTMs introduce recursive connections between hidden layer activations. A forward pass of an LSTM unit is summarized by the following equations,

$$f_t = \sigma_g(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{9}$$

$$i_t = \sigma_g(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{10}$$

$$o_t = \sigma_g(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{11}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot (\tanh(W_c[h_{t-1}, x_t] + b_c)) \tag{12}$$

$$h_t = o_t \odot \tanh(c_t) \tag{13}$$

where W and b represent the weights and the bias associated with the indexed layer, respectively, whereas σ_g is the sigmoid activation function. The LSTM cell state is represented by c and the hidden units by h.

D. *Results and Discussion*

In this section, we will discuss the *difference* in performance between our proposed model and the baselines independently. The results show that our proposed model outperforms the gradient-boosted decision trees as well as recurrent neural network in more than 60% of the design parameters, with an average F1 score improvement of 6.41% and 1.31% respectively. Figures 5 represents the *difference* in performance between the proposed convolutional model and GBDT across two different time steps, 48 time steps (4 hours) and 96 time steps (8 hours) for the aforementioned thresholds.

We notice that the difference in F1-score, which defines the harmonic mean between precision and recall, increases as the horizon increases. The performance scales similarly in both datasets. Although GBDT demonstrates competitive performance with horizons up to 15 minutes, i.e. 6 time steps, for longer horizons, its performance falls around ∼20%. We also notice that the immediate history, represented with the 4 hours, performs better than an 8 hour time frame. This can be expected, since in parking, more immediate behavior can be more informative than distant behavior, due to the dynamic nature of traffic.

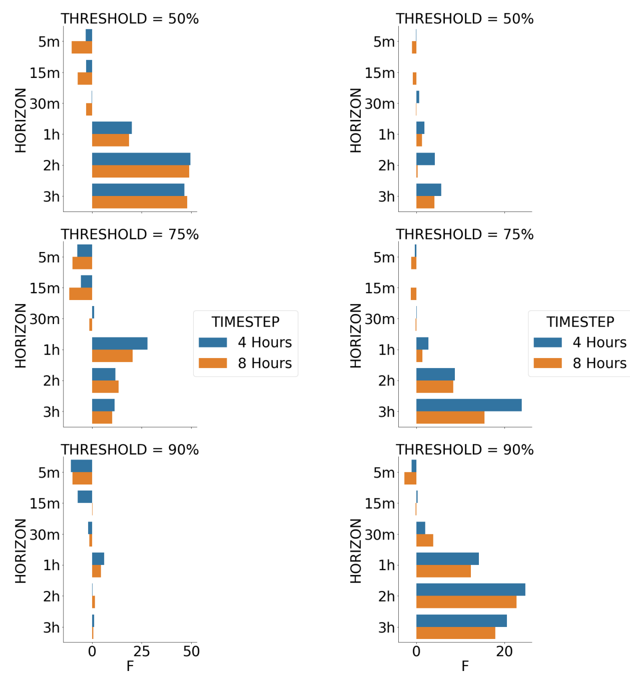
For the off-street parking data, provided by BANES dataset, we notice that as the cut-off threshold increases, our model demonstrates higher performance, unlike GBDT which deteriorates as the data becomes sparser. On the other hand, for on-street parking, provided by the PbP dataset, our proposed model and GBDT fare similarly with long horizons and sparser data.

If we focus on recall, Table I, which in this scenario, represents the percentage of predicted occupied spots that were indeed occupied, we see that as the data becomes sparser for on-street parking, the proposed model and GBDT demonstrate similar behavior, however for off-street parking, our method clearly beats GBDT. The more immediate history results in overall better performance as well, the blue column is always higher than the orange column.

Finally by investigating the precision, Figure 6, which in this scenario, represents the percentage of predicted free spots that were indeed free, our method outperforms GBDT in most of the off-street parking settings. We also notice that for on-street parking, the precision of our improves in comparison as the horizon increases.

TABLE I: Average difference in Recall with GBDT

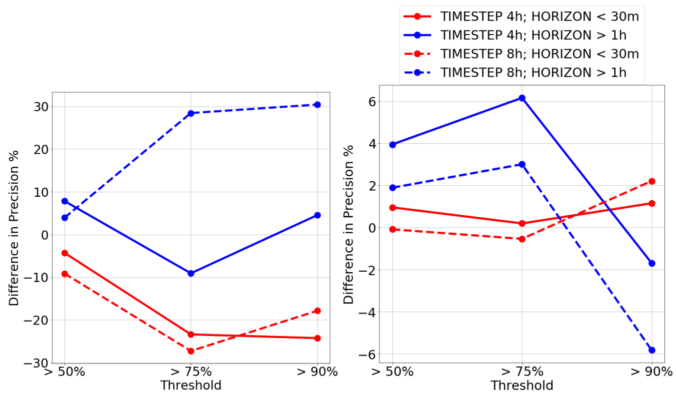
Horizon	Timesteps	Threshold	On-street	Off-street
Less than 30 Minutes	4 Hours	50%	0.03	-0.68
Less than 30 Minutes	8 Hours	50%	-4.56	-1.25
Less than 30 Minutes	4 Hours	75%	7.10	-0.41
Less than 30 Minutes	8 Hours	75%	3.79	-1.32
Less than 30 Minutes	4 Hours	90%	2.51	-0.42
Less than 30 Minutes	8 Hours	90%	1.90	-1.57
More than 1 Hour	4 Hours	50%	36.31	3.72
More than 1 Hour	8 Hours	50%	36.37	1.84
More than 1 Hour	4 Hours	75%	13.46	14.45
More than 1 Hour	8 Hours	75%	11.72	11.52
More than 1 Hour	4 Hours	90%	2.42	23.73
More than 1 Hour	8 Hours	90%	1.59	23.00



(a) On-street (b) Off-street

Fig. 5: Difference in F1-score with GBDT

Moving the discussion to the *difference* between the proposed model and the LSTM with 128 hidden units, Figure 7, clearly reflects that the proposed model behaves differently as a function of sparsity and horizon. In on-street parking data, with a threshold of 50%, we outperform LSTMs on longer horizons, whereas for a threshold of 90%, the proposed model is better at handling sparser data. For off-street parking, our



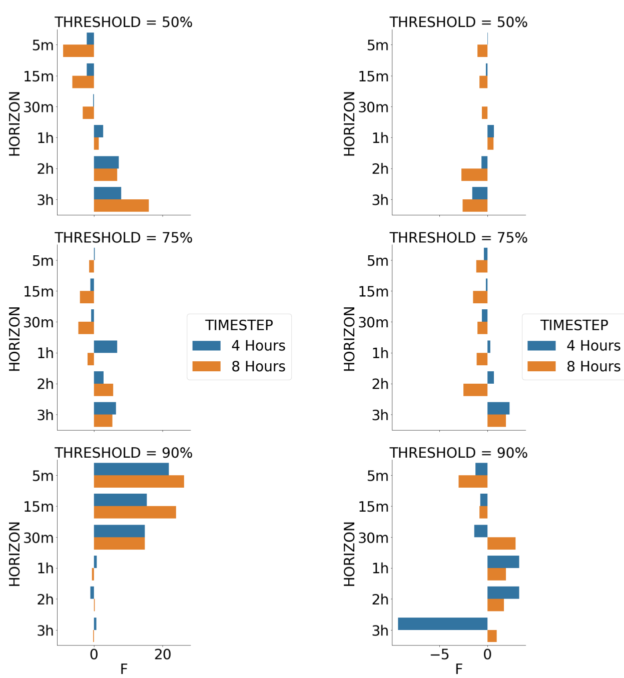
(a) On-street (b) Off-street

Fig. 6: Average difference in Precision with GBDT

TABLE II: Average difference in Recall with LSTM

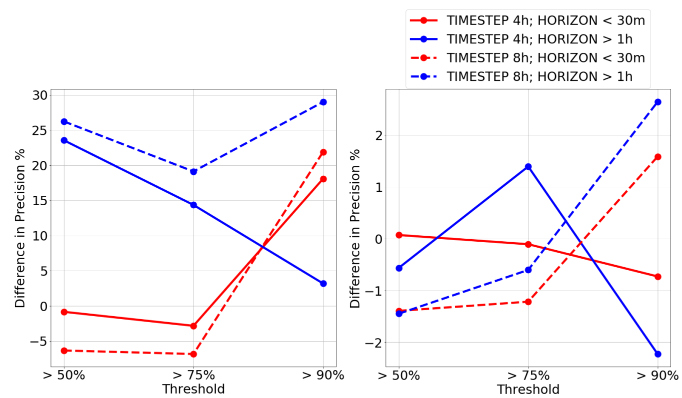
Horizon	Timesteps	Threshold	On-street	Off-street
Less than 30 Minutes	4 Hours	50%	-2.24	-0.16
Less than 30 Minutes	8 Hours	50%	-6.11	-0.28
Less than 30 Minutes	4 Hours	75%	1.83	-0.67
Less than 30 Minutes	8 Hours	75%	0.11	-1.30
Less than 30 Minutes	4 Hours	90%	14.86	-1.54
Less than 30 Minutes	8 Hours	90%	18.48	-2.15
More than 1 Hour	4 Hours	50%	-15.08	-0.50
More than 1 Hour	8 Hours	50%	-18.58	-1.70
More than 1 Hour	4 Hours	75%	-7.74	0.68
More than 1 Hour	8 Hours	75%	-8.19	-0.51
More than 1 Hour	4 Hours	90%	-44.06	0.47
More than 1 Hour	8 Hours	90%	-15.62	0.90

approach demonstrates competitive performance, especially at longer horizons, with an average variation in F1-score of $\sim$3%. From Figure 8 and Table II, we realize that LSTMs are inclined to predict more positives, i.e. occupied positions, whereas our proposed model is more conservative. This assumption is based on the results that highlight a higher difference in precision as compared to a lower difference in recall. Overall, our proposed approach is better suited at handling sparser data for longer horizons.



(a) On-street (b) Off-street

Fig. 7: Difference in F1-score with LSTM



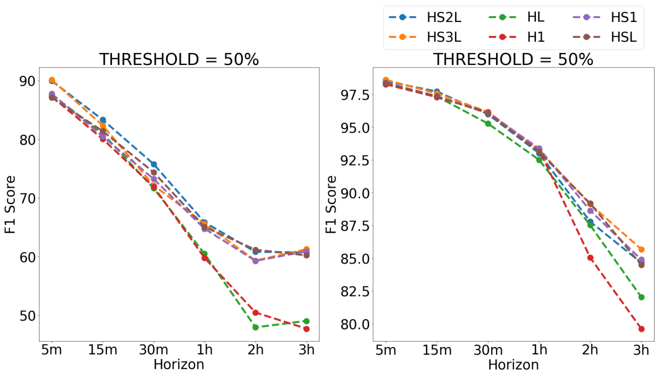
(a) On-street (b) Off-street

Fig. 8: Average difference in Precision with LSTM

E. Ablation Study

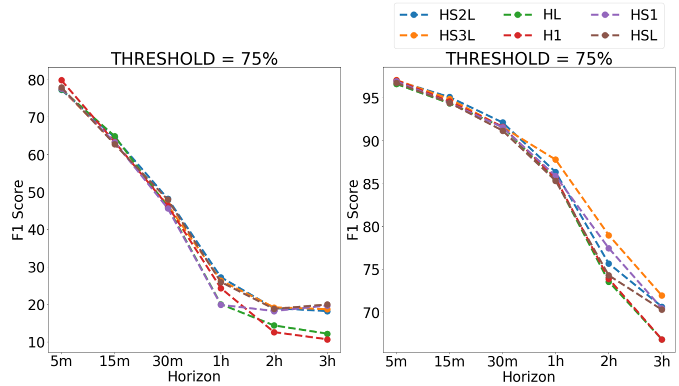
We conducted an ablation study to highlight the importance of statistical features and location identifiers in modeling the temporal dependencies of parking data. We use the following models: (a) univariate time-series without any statistical or location data, H1, (b) univariate time-series with location identifier, HL, (c) univariate time-series with statistics of the point at the horizon, HS1, (d) univariate time-series with both statistics of the target point and the location identifier, HSL, (e) multivariate time-series, where the statistics are appended as additional channels to the occupancy distribution, plus the location identifier, HS2L, and finally, (f) multivariate time-series with the location identifier and the statistics of the target point, HS3L. Figures 9, 10, and 11 represent the average F1-score of on- and off-street parking for multiple horizons across the two time-windows tested.

Immediately we notice that as the horizon increases, the performance of all the models decreases. At first, all models perform relatively similar, with F1-scores above $\sim$90%. It is not until the horizon exceeds one-hour that we start noticing



(a) On-street (b) Off-street

Fig. 9: Ablation study for a threshold of 50%



(a) On-street (b) Off-street

Fig. 10: Ablation study for a threshold of 75%

a considerable difference in performance.

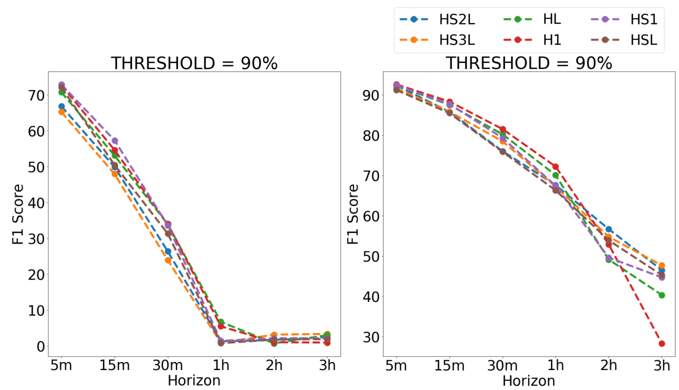
Appending a location identifier to the convolutional model consistently achieves better results. This comes as no surprise since parking locations are unique, and every parking spot follows a specific temporal behavior depending on its geo-location, number of spots, neighboring parking locations, etc. Hence, augmenting the latent temporal representation with location identifiers introduces a bias in the activation, and boosts the performance of the entire model.

Appending statistical features improves the overall performance as well. As mentioned earlier, the statistics include minimum, maximum, and variance of the bins, as well as one-hot encoded time features of the target point. These statistics capture the seasonalities and trends that are present in parking behavior. Using statistics of the target input, without additional information, HS1, outperforms location identifiers. This might be the case due to the fact that these statistics provide a meaningful insight into the target point, as opposed to the location itself, which gives insight to the specific parking location as whole.Target statitstics are significantly better when as the data becomes sparser.

We notice finally that augmenting all of the statistical with location identifiers, HS3L, achieves the best amongst all variations. This multi-modal feature fusion introduces more diversity into the data, and allows the model to capture the dynamics of the data on both spatial aspect and the local temporal aspect.

VI. Conclusion

Parking availability prediction is an emerging topic that is gaining more and more interest among companies with access to such traffic data. Conventional time-series forecasting approaches result in good performances for immediate horizons, however, they are not capable of maintaining high F1-score across all horizons. In this paper, we propose a new hybrid convolutional approach that outperforms time-series forecasting benchmarks, namely gradient-boosted decision trees and recurrent neural networks for more than 60% of the design parameters, with a respective average improvement of 6.41%



(a) On-street (b) Off-street

Fig. 11: Ablation study for a threshold of 90%

and 1.31%. We show that our model is capable of handling sparse data, with less than 1% of the data being occupied, as is evident by the difference in the F1-score. The hybrid approach, which builds on top of the convolutional layer by appending location and/or statistical features, improves the performance, especially at longer horizons.

The training is agnostic to any type of external data that might effect the traffic behavior, such as the weather or a local event. Future work will focus on data augmentation, and to improving the convolutional feaure extractor. It would also be interesting to investigate the effect of limiting the horizondata the training data to the transcations are registered. This would drastically reduce the training data, however it would be more realistic as it will be trainined to predict for horizons that are based on real customer preferences.

REFERENCES

[1] Greengard, Samuel. ”Between the lines.” Communications of the ACM 58.6 (2015): 15-17.

[2] Mathur, Suhas, et al. ”Parknet: drive-by sensing of road-side parking statistics.” Proceedings of the 8th international conference on Mobile systems, applications, and services. ACM, 2010.

[3] Shin, Jong-Ho, and Hong-Bae Jun. ”A study on smart parking guidance algorithm.” Transportation Research Part C: Emerging Technologies 44 (2014): 299-317.

[4] Yang, Zhaosheng, Honghong Liu, and Xinyue Wang. ”The research on the key technologies for improving efficiency of parking guidance system.” Intelligent Transportation Systems, 2003. Proceedings. 2003 IEEE. Vol. 2. IEEE, 2003.

[5] Kimms, Alf, K-C. Maassen, and Susanne Pottbcker. ”Guiding traffic in the case of big events with spot checks on traffic and additional parking space requirements.” Central European Journal of Operations Research 20.4 (2012): 755-773.

[6] Bates, John, and David Leibling. ”Spaced Out: Perspectives on parking policy.” (2012).

[7] Inci, Eren. ”A review of the economics of parking.” Economics of Transportation 4.1-2 (2015): 50-63.

[8] Giuffr, Tullio, Sabato Marco Siniscalchi, and Giovanni Tesoriere. ”A novel architecture of parking management for smart cities.” Procedia-Social and Behavioral Sciences 53 (2012): 16-28.

[9] Revathi, G., and VR Sarma Dhulipala. ”Smart parking systems and sensors: A survey.” Computing, Communication and Applications (ICCCA), 2012 International Conference on. IEEE, 2012.

[10] Nandugudi, Anandatirtha, et al. ”PocketParker: pocketsourcing parking lot availability.” Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing. ACM, 2014.

[11] Mathew, Sujith Samuel, et al. ”Building sustainable parking lots with the Web of Things.” Personal and ubiquitous computing 18.4 (2014): 895-907.

[12] Shinde, Snehal, et al. ”Prediction of Parking Availability in Car Parks using Sensors and IOT: SPS.” International Journal of Engineering Science and Computing 6 (2016).

[13] Vlahogianni, Eleni I., et al. ”A real-time parking prediction system for smart cities.” Journal of Intelligent Transportation Systems 20.2 (2016): 192-204.

[14] Zheng, Yanxu, Sutharshan Rajasegarar, and Christopher Leckie. ”Parking availability prediction for sensor-enabled car parks in smart cities.” Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2015 IEEE Tenth International Conference on. IEEE, 2015.

[15] Pflgler, Christoph, et al. ”Predicting the availability of parking spaces with publicly available data.” Informatik 2016 (2016).

[16] Rajabioun, Tooraj, and Petros A. Ioannou. ”On-street and off-street parking availability prediction using multivariate spatiotemporal models.” IEEE Transactions on Intelligent Transportation Systems 16.5 (2015): 2913-2924.

[17] Chen, Xiao. ”Parking occupancy prediction and pattern analysis.” Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, Tech. Rep. CS229-2014 (2014).

[18] Lin, Trista, et al. ”A survey of smart parking solutions.” parking 1524 (2017): 9050.

[19] Richter, Felix, Sergio Di Martino, and Dirk C. Mattfeld. ”Temporal and spatial clustering for a parking prediction service.” Tools with Artificial Intelligence (ICTAI), 2014 IEEE 26th International Conference on. IEEE, 2014.

[20] Caliskan, Murat, et al. ”Predicting parking lot occupancy in vehicular ad hoc networks.” Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th. IEEE, 2007.

[21] Klappenecker, Andreas, Hyunyoung Lee, and Jennifer L. Welch. ”Finding available parking spaces made easy.” Ad Hoc Networks 12 (2014): 243-249.

[22] Zheng, Yanxu, et al. ”Smart car parking: temporal clustering and anomaly detection in urban car parking.” Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2014 IEEE Ninth International Conference on. IEEE, 2014.

[23] Rong, Yuecheng, et al. ”Du-Parking: Spatio-Temporal Big Data Tells You Realtime Parking Availability.” Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018.

[24] Zhang, Junbo, et al. ”DNN-based prediction model for spatio-temporal data.” Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. ACM, 2016.

[25] Yu, Haiyang, et al. ”Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks.” Sensors 17.7 (2017): 1501.

[26] Li, Yaguang, et al. ”Diffusion convolutional recurrent neural network: Data-driven traffic forecasting.” (2018).

[27] Tobler, Waldo R. ”A computer movie simulating urban growth in the Detroit region.” Economic geography 46.sup1 (1970): 234-240.

[28] Kingma, Diederik P., and Jimmy Ba. ”Adam: A method for stochastic optimization.” arXiv preprint arXiv:1412.6980 (2014).

[29] Chen, Tianqi, and Carlos Guestrin. ”Xgboost: A scalable tree boosting system.” Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. ACM, 2016.