Time-to-Green predictions

A framework to enhance SPaT messages using machine learning

Conference Paper

Author(s):
Genser, Alexander; Ambühl, Lukas; Yang, Kaidi; Menendez, Monica; Kouvelas, Anastasios

Publication date:
2020

Permanent link:
https://doi.org/10.3929/ethz-b-000443743

Originally published in:
https://doi.org/10.1109/ITSC45102.2020.9294548

Time-to-Green predictions: A framework to enhance SPaT messages using machine learning

Alexander Genser[1], Lukas Ambühl[1], Kaidi Yang[2], Monica Menendez[3] and Anastasios Kouvelas[1]

Abstract— **Recently, efforts were made to standardize Signal Phase and Timing (SPaT) messages. Such messages contain the current signal phase with a prediction for the corresponding residual time for all approaches of a signalized intersection. Hence, the information can be utilized for the motion planning of human-driven/autonomously operated individual or public transport vehicles. Consequently, this leads to a more homogeneous traffic flow and a smoother speed profile. Unfortunately, adaptive signal control systems make it difficult to predict the SPaT information accurately. In this paper, we propose a novel machine learning approach to forecast the time series of residual times. A prediction framework that utilizes a Random Survival Forest (RSF) and a Long-Short-Term-Memory (LSTM) neural network is implemented. The machine learning models are compared to a Linear Regression (LR) model. For a proof of concept, the models are applied to a case study in the city of Zurich. Results show that the machine learning models outperform the LR approach, and in particular, the LSTM neural network is a promising tool for the enhancement of SPaT messages.**

I. INTRODUCTION

Digitalization has changed the transportation domain significantly in the last decade. The availability of several new data sources (i.e., sensor technology or vehicle technology) allows for data-driven methodologies that can be incorporated into well-established traffic management systems on a macro- and micro-scopic level. Furthermore, upcoming developments, such as Vehicle-to-Infrastructure (V2I) communication, open the door for new approaches that allow considering communication between vehicles and infrastructure. Recent evolution in traffic signal control of urban intersections (e.g., actuated signal control, self-control algorithms, etc.) influence the signal phases and result in variable green, red and cycle times. Hence, speed advisory systems would benefit from the information about when the next green phase starts so that vehicles do not have to stop when crossing an intersection. Such information is provided in Signal Phase and Timing (SPaT) messages. Unfortunately, predictions for residual times of these quantities are not a trivial reverse engineering and still provide a burden for field applications of SPaT broadcasts. Hence, a sophisticated modeling approach for accurate predictions is required and still open to research.

[1]Alexander Genser, Lukas Ambühl, and Anastasios Kouvelas are with the Institute for Transport Planning and Systems, ETH Zurich, Switzerland. {alexander.genser,lukas.abmuehl,tasos.kouvelas}@ivt.baug.ethz.ch

[2]Kaidi Yang is with the Autonomous Systems Lab, Stanford University, USA. kaidi.yang@stanford.edu

[3]Monica Menendez is with the Division of Engineering, NYU Abu Dhabi, UAE. monica.menendez@nyu.edu

In this paper we propose a framework with two Machine Learning (ML) approaches to predict the residual time of each phase of an intersection. To capture the nonlinear relationship between the signal information and the multiple detectors of an intersection, we introduce the problem as a time series forecast and apply a Random Survival Forest (RSF) and a Long-Short-Term-Memory (LSTM) neural network. The novel approaches are compared to a simple ML approach, i.e., a Linear Regression (LR). To prove the concept, historical Loop Detector (LD) and signal timing data from an intersection in the city of Zurich is utilized. The area under investigation includes signal priority for public transportation (i.e., signal priorities change the control behavior of the intersection irregularly).

The remainder of this paper is organized as follows: An overview of recent research on the prediction of residual times is provided in Section II. In Section III the time series problem and the Time-to-Green (T2G) prediction framework are introduced. Furthermore, the theory of the utilized models and performance metrics are given in Section III-C to III-F. Finally, the applicability of all models is shown in a case study with a detailed presentation of prediction results in Section IV. A conclusion and proposal of future work are given in Section V.

II. BACKGROUND

Recently, efforts were made to standardize SPaT messages. Such messages contain the current phase with a prediction for the corresponding residual time for all approaches of a signalized intersection. Hence, SPaT information allows a more efficient and environmentally friendly motion planning of human-driven and/or autonomously operated individual or public transport vehicles. Especially in urban areas, this would lead to more homogeneous traffic flow and a smoother speed profile (i.e., the absence of speeding and heavy breaking between traffic lights). Most existing speed advisory systems rely on SPaT information [1], [2], which includes elements such as the start time of a signal phase, phase duration, or the next time this signal phase starts. Although it is widely accepted that broadcasting SPaT information is potentially beneficial for traffic systems [3], SPaT information is rarely provided in reality. One challenge lies in the accurate prediction of SPaT information. Modern traffic control systems (e.g., in the city of Zurich) are typically adaptive to vehicle demand, resulting in non-repetitive control sequences. Although adaptive control systems can efficiently prioritize public transport and serve car demand, it is difficult to predict SPaT information for such systems accurately.

Most existing works develop methods to obtain SPaT information for pre-timed traffic signals based on aggregated trajectory data. In such works, signal timings are unknown (either fixed or change slowly with time) and estimated using traffic models [4]–[7] or ML approaches [8], [9]. For example, [4] and [5] employed a queue discharging model to estimate the start of green signals based on aggregated low-frequency bus and probe data. Such methods typically rely on the underlying assumption that cycle length is fixed, with only few exceptions (e.g., [5] and [9]) that are able to identify the occasional changes in the traffic signal timing plan. Moreover, these works are based on the aggregation of historical vehicle trajectories, assuming that the historical signal timings are unknown.

Some other studies in the literature propose probabilistic methods to predict SPaT information such as [10]. The study estimates the conditional distribution of each signal phase given real-time measurements to predict the phase duration as a conditional expectation and a confidence interval. Nevertheless, this latter approach requires separate aggregations for different cycle lengths.

All the mentioned works are based on (a) vehicle trajectory data, or (b) historical signal data, and do not incorporate vehicle detection into the modeling. However, in a complex signalized intersection with multiple approaches and movements, the signal timing can be determined jointly by many LDs. Also, it is important to consider the temporal relation between the traffic signals and the LDs because a delay between the LD activation and the change of signal state could be apparent. Therefore, the tuning of a model that can identify the nonlinear relationship and incorporate both data sources to estimate the residual time is promising and still open to research.

III. METHODOLOGY

We approach the prediction of residual times as a time series problem and incorporate the definition into our proposed T2G framework (Section III-A and III-B) where the baseline model (LR) and the ML models (RSF and LSTM) are tested. Section III-C and III-D provide the theory of the utilized models. The performance assessment is modeled with the metrics introduced in Section III-F.

A. Problem definition

The prediction of the residual time of a signal phase (i.e., the T2G) is based on signal and LD data. Let i be a signal device where $i \in \mathcal{S}$ and j an LD device where $j \in \mathcal{D}$. $\mathcal{S}$ and $\mathcal{D}$ are the sets of signals and LDs, respectively. Consequently, let $s_i(t)$ be the state of a signal i at time t defined as follows:

$$s_i(t) = \begin{cases} 0, & \text{if } s_i \text{ is red} \\ 1, & \text{otherwise.} \end{cases} \tag{1}$$

Analogously, we define the state of an LD j at t with the function $d_j(t)$.

$$d_j(t) = \begin{cases} 0, & \text{if } d_j \text{ is not occupied} \\ 1, & \text{otherwise.} \end{cases} \tag{2}$$

Considering (1) and (2) note that both inputs are defined as categorical variables. We introduce T2G prediction as a time-series forecast and construct the problem with historical data of signal i. $c_i(t)$ denotes the current cycle signal i is operating. $c_i \in \mathcal{C}$ where $\mathcal{C}$ is the set of possible cycles for i.

Furthermore, we need to determine the total length of red time of the signal cycle $c_i(t)$, denoted as $r_i(c_i(t))$ and the current time step in a cycle c as $q_i(c_i(t))$. Finally, let $y_i(t)$ be the T2G of a signal i formulated as follows:

$$y_i(t) = \begin{cases} r_i(c_i(t)) - q_i(c_i(t)), & \text{if } s_i(t) = 0 \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

The proposed formulation can be utilized to determine the T2G $y_i(t)$ for a signal i in every cycle c_i when a traffic light is red (i.e, $s_i(t) = 0$). During green ($s_i(t) = 1$), the T2G remains 0. Consequently, a linear time series is constructed as denoted in (3).

B. T2G framework

In the following, we introduce a T2G prediction framework that allows a generic application to any intersection, whose architecture is depicted in Figure 1. The denoted blocks (1)–(3) follow the general working steps of a supervised ML procedure. The raw data (i.e., LD and signal data from the traffic operator) functions as an input to the data preprocessing (Block (1)). Within this step, the data cleaning, data aggregation and transformation take place. The outputs are the pre-defined quantities $s_i(t)$, $d_j(t)$ and $y_i(t)$. Note that $r_i(c_i(t))$ and $q_i(c_i(t))$ only represent internal variables for determining $y_i(t)$. Therefore, these quantities are excluded in the output of Block (1). The outputs are summarized in the set $\mathcal{H}$ and serve as an input to Block (2), where the feature engineering is performed. Here, the most relevant input variables are determined by eliminating correlations and – if it is a model requirement – variable transformations are computed.

In Block (3) the input is split into training and test data set (70% and 30% of the data, respectively). Consequently, a set of models $\mathcal{M}$ is considered, which is specified by the user. After training and testing all elements of $\mathcal{M}$ the best model is selected and predictions for the set of traffic signals are generated. Note that the process of tuning a proper prediction model requires a number of iterations between Block (2) and (3). In this paper, we define an LR, an RSF and an LSTM model as the members of $\mathcal{M}$. First, the baseline model (i.e., LR) is introduced. Afterwards, we introduce the RSF and LSTM model.

C. Baseline model

We introduce the LR which can be defined as follows:

$$\begin{aligned} \hat{y}_{i,o} = \beta_{i,0} + \beta_{i,1}x_{o,1} + \beta_{i,2}x_{o,2} + ... \\ +\beta_{i,p}x_{o,p} + E_{i,o}, \ \forall o = 1,..,T, \end{aligned} \tag{4}$$

where $\hat{y}_{i,o}$ is the T2G (response variable) for signal i and observation o, $\beta_{i,0}$ represents the intercept term and $\beta_{i,1}$ to $\beta_{i,p}$ are the regression coefficients for the p predictors $x_{o,1}$ to $x_{o,p}$ (i.e., the members of the final LD and signal

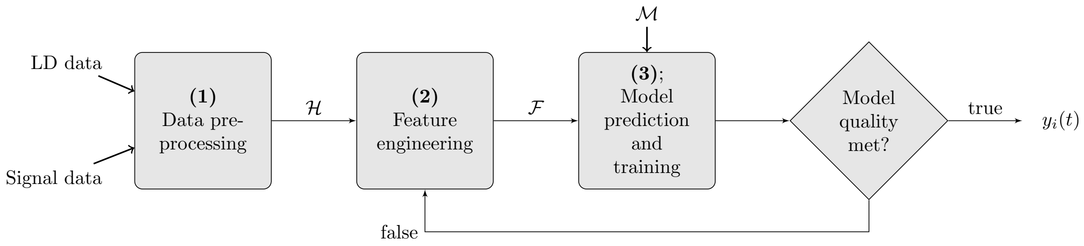

Fig. 1: T2G framework. The input data is represented by LD and signal data from the traffic operator, respectively.

Algorithm 1 RSF pseudo code

procedure DORSF
  $\mathcal{B} \leftarrow$ Bootstrap samples from the original training data set
  $\mathcal{T} \leftarrow$ Grow a survival tree $\forall s \in \mathcal{B}$, where s is a sample from $\mathcal{B}$
  $\mathcal{O} \leftarrow$ Exclude out-of-bag data from all $s \in \mathcal{B}$
  Grow tree to full size with the constraint that a leaf node $l \in \mathcal{L}$ has no less than $d_0 > 0$ unique deaths; $\mathcal{L}$ is a set of leaf nodes
  $\mathcal{C} \leftarrow$ Calculate the Cumulative Hazard Function (CHF) $\forall t \in \mathcal{T}$, where t is a survival tree for one bootstrap sample
  $\mathcal{P} \leftarrow$ Calculate prediction error using the set $\mathcal{O}$

feature set $\mathcal{F}$ described above), respectively. The error term is denoted by $E_{i,o}$ and follows a Gaussian distribution (i.e., $E_{i,o} \sim \mathcal{N}(0, \sigma_{E_{i,o}})$); T denotes the prediction horizon. The solution for $\hat{y}_{i,o}$ is found by applying the Ordinary Least Square (OLS) method. The fitted model can be used to determine a prediction of the T2G for all given traffic signals by obtaining the conditional expected value of the response. To obtain the LR model, the build-in R-package was utilized. Research that similarly introduces LR models within this context can be found in e.g., [11].

D. Machine learning models

To assess the performance of ML models for T2G predictions, two models are introduced. As a first model (and second member of $\mathcal{M}$), an RSF is utilized. The RSF method is based on the concept of survival analysis, i.e., the analysis of time duration until a certain event occurs. Our constructed time series can be analyzed in such a way as it is expected that after the T2G the traffic light switches to green, meaning the time series is 0 for a certain time horizon. Therefore, we utilize the R-package *randomForestSRC*, which implements the pseudo-code presented in Algorithm 1, proposed by [12]. For the detailed mathematical background of RSF, the interested reader is referred to [13].

The second ML model incorporated into the T2G framework is a LSTM network which is a special type of Recurrent Neural Networks (RNN). To address the drawbacks of standard memory-less RNNs (vanishing gradient or exploding), extensions regarding the network architecture with a memory block was proposed. Along with other neural network designs, an LSTM is constructed with an input, hidden and output layer. The hidden layer is designed with a so called memory block, containing memory cells. The state of these cells is influenced by memorizing the temporal state and gating units that control the information flow in one memory cell. In addition, input and output gates are implemented to control the input and output activation's, respectively. When the information state of a memory cell is outdated, a forget gate allows an automatic reset to forget information that loses importance while evolving in time [14]. The model formulation is denoted with an input $x = (x_{i,1}, x_{i,2}, ..., x_{i,T})$ and the output $\hat{y} = (\hat{y}_{i,1}, \hat{y}_{i,2}, ...\hat{y}_{i,T})$. The vector X holds all the input features from $\mathcal{F}$. Again, $\hat{y}$ is the predicted response and T is the prediction horizon. To predict the T2G in the next time step, the following equations are introduced (for simplicity, note that the index for signal i is omitted):

$$a_t = \mathrm{sig}\Big(W_{ax}x_t + W_{am}m_{t-1} + W_{ac}c_{t-1} + b_a\Big), \quad (5)$$

$$f_t = \mathrm{sig}\Big(W_{fx}x_t + W_{fm}m_{t-1} + W_{fc}c_{t-1} + b_f\Big), \quad (6)$$

$$c_t = f_t \bullet c_{t-1} + a_t \bullet g\Big(W_{cx}x_t + W_{cm}m_{t-1} + b_c\Big), \quad (7)$$

$$o_t = \mathrm{sig}\Big(W_{ox}x_t + W_{om}m_{t-1} + W_{oc}c_t + b_o\Big), \quad (8)$$

$$m_t = o_t \bullet h(c_t), \quad (9)$$

$$\hat{y}_t = W_{\hat{y}m}m_t + b_{\hat{y}}, \quad (10)$$

where a_t, f_t, c_t, o_t and m_t are the states of the input gate, forget gate, cell state, output gate and memory gate, respectively. The variables W and b denote the weight matrices and bias vectors, respectively and are utilized to connect input, hidden and output layer. Note that $\mathrm{sig}(\cdot)$ defines the logistic function (i.e., sigmoid function); $g(\cdot)$ and $h(\cdot)$ are logistic functions with intervals $[-2, 2]$ and $[-1, 1]$, respectively [14]. The work in [15] introduces similar mathematical descriptions of LSTM networks. The implementation in our framework is performed with TensorFlow and Keras.

E. Moving window procedure

To improve the prediction quality we apply a moving window procedure to the input of the LR and RSF model. Let w be a moving time window in seconds and the matrix I our input matrix which represents all elements of $\mathcal{F}$. The dimension of I is denoted by $\dim(m_I, n_I)$. To apply our moving window procedure we introduce a transformed matrix J with $\dim(m_I - w, n_I - w)$. Thereby, the first dimension $m_I - w$ results from the end of the time series as we only consider complete windows. We fill the values of J row by row, each row corresponding the concatenated feature vectors of the corresponding w time steps from I. This procedure improves the results significantly as the training process provides inputs at time step t for multiple time instances. Note that according to the model definitions of LSTM a moving window procedure is not applicable.

F. Performance metrics

After training the model, the testing is performed with the test data set. For the error value computation, the Mean-Absolute-Error (MAE) and the Root Mean Square Error (RMSE) are utilized. The performance metrics are introduced by (11) and (12):

$$\mathrm{MAE} = \frac{1}{T}\sum_{k=1}^{T}\left|\hat{y}_{i,k} - y_{i,k}\right|, \tag{11}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{T}\sum_{k=1}^{T}\left|\hat{y}_{i,k} - y_{i,k}\right|^{2}}. \tag{12}$$

$\hat{y}_{i,k}$ again represents the estimated T2G of signal i and $y_{i,k}$ is the T2G from the test data set. k is here utilized to sum the errors over T. As the prediction is only applied when the T2G is decreasing (i.e., when the signal is red), only those parts of the signals $\hat{y}_{i,k}$ and $y_{i,k}$ are considered for computation of MAE and RMSE. We obtain these parts automatically by considering the derivative of $y_{i,k}(t)$.

In the following section, we show the application of the T2G framework to a test intersection in the city of Zurich.

IV. CASE STUDY

The analysis is based on a data set from an intersection in the city center of Zurich. From north to south and vice versa, there are tram lines that are prioritized by the signal control. Figure 2 depicts the intersection with the 10 associated LDs (indicated as rectangles at the intersection approaches) and 12 traffic signals (indicated by the circled numbers). The tram lines are indicated by the red dashed lines. Note that traffic signal 3 is for bicycles which are allowed to go straight ahead. No separate detector data is available for this signal. Hence, its phase corresponds to the one of signal 2. Signal 11 is installed for the tram to indicate potentially leaving the stop, but Signal 6 remains crucial at this approach. 15 days of high-resolution data from January 2019 are available. The intersection is operated with an actuated control with no fixed cycle times. Therefore, it is evident that (a) a prediction of the T2G is not a trivial reverse engineering, and (b) the variances give an indication that simple prediction approaches might fail to make a prediction that satisfies accuracy requirements.

A. Descriptive and correlation analysis

We prove our concept by obtaining a subset of the 15 days' peak hours out of the complete data set.

As the potential correlation between variables degrades the model quality, we investigate all LDs and signal devices (Figure 3). The data shows almost no correlation between LD

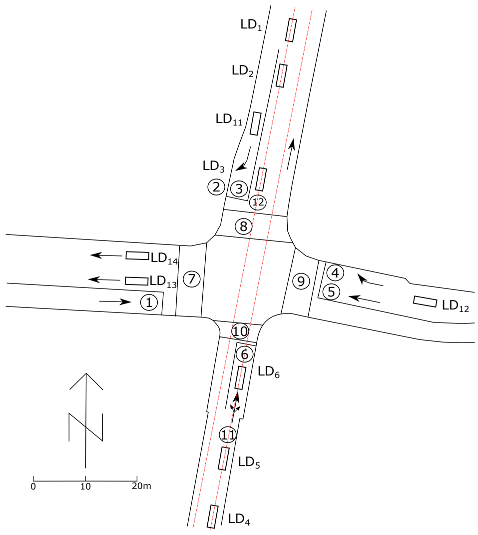

Fig. 2: Test intersection in the city center of Zurich, CH.

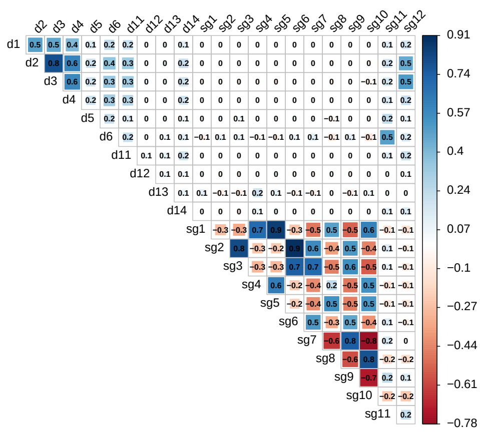

Fig. 3: Correlations of all LD (d) and signal devices (sg).

devices (except LD 2 and LD3) and between specific LD and signal devices. A more detailed analysis of the signal devices depict some significant positive correlations (e.g., signals 1 and 5 or signals 2 and 6). Nevertheless, this is expected as these signals control compatible traffic streams (see Figure 2). The same argument holds for the signals regulating pedestrian streams (signal pairs (7,9) and (8, 10)).

The correlation between signals 2 and 3 might occur because the two streams get green at the same time, but the green time for bicycles is smaller as a longer clearing time is required. Consequently, we remove one of the variables that are members of correlated pairs. Note that for this analysis we decide to keep all variables with a correlation coefficient smaller than 0.8. The final signal members of the feature set $\mathcal{F}$ are $i = \{3, 4, 5, 6, 7, 8, 11, 12\}$. For the LDs the devices $j = \{1, 3, 4, 5, 6, 11, 12, 13, 14\}$ are utilized. Consequently, 17 features are used as input LR, RSF, and LSTM models.

B. Results

In this section, we apply our set of models $\mathcal{M}$ to the training and testing procedure. We asses the model quality by utilizing a subset of the data (7:00am to 10:00am of one day) and split the data to 70% train and 30% test data, respectively. We present the model outputs for signal $i = 1$. First, the LR model is applied with standard settings and no variable transformations. Nevertheless, the general assumptions required to justify the application of OLS are considered in the analysis. A window size of $w = 10$ is chosen. The model shows that 92 out of 170 parameters are statistically significant on the 95% level and the adjusted R^2 is determined to 0.89.

The RSF is applied with a number of trees $n = 100$; n is chosen considering the trade-off between gain in improvement versus the increase in computational time. Furthermore, the data is centered and scaled to obtain better model performance. Again, the window size is chosen with $w = 10$. Analyzing the model output shows that the RSF model explains 95.62% of the training data's variance with an average number of terminal nodes of 11951.28. The LSTM model was trained with 50 epochs and a batch size of 2000. The architecture is specified with 50 hidden layers and the first layer is designed according to the shape of the input data. To ensure a valid model quality, the loss function (MAE) was analyzed to prevent the model from overfitting the training data. To allow a performance comparison between the models, we test all members of $\mathcal{M}$ on the test data set and show the $\overline{\mathrm{MAE}}$ for all calculated errors and the corresponding standard deviation σ_{MAE} in Table I.

The results show that the LR model provides $\overline{\mathrm{MAE}}$ values between 2.13 and 2.66 sec for traffic lights that regulate vehicle traffic streams (signals 1 to 10). For public transport signals, the prediction accuracy shows a high error. This is because the model is not able to capture the signal peaks that occur when there is no demand on these streams. An example of this behavior is depicted in Figure 4 (signal peak at 11:00am). Besides, the linear model suffers from negative values during the period that T2G is zero and is also likely to miss such signal patterns, which result in a constant T2G greater than zero (Figure 5 11:02am).

The RSF model demonstrates the second best performance with $\overline{\mathrm{MAE}}$ ranging from 0.64 to 1.37 sec for signals 1–10, with a corresponding low variance σ_{MAE}. For signal 11 and signal 12, the model also fails to capture high signal peaks accurately. Nevertheless, the RSF prediction never misses a green phase (when T2G is zero). A sample prediction is shown in Figure 5 with the described behavior. Our best candidate is the LSTM neural network where the condition $\overline{MAE} < 1$ holds for all signals. The highest error value can be depicted for signal 11 with an $\overline{MAE}$ of 0.42 and σ_{MAE} of 0.03. This corresponds to an $\overline{RMSE}$ of 0.43 with σ_{MAE} of 0.02. Figure 6 shows the test and prediction data with an accurate fit of the time series. Therefore, an LSTM should be utilized as a generic and flexible tool for predicting the T2G series of a traffic signal. Compared to the RSF the error values decrease significantly and the model is able to capture more irregular patterns for traffic lights of public transport. In addition, the LSTM meets the accuracy requirement for T2G predictions.

Finally, we utilize our best candidate (LSTM) and test the trained model in a peak-hour scenario of another weekday. We determine 10 minutes of historical data and test the prediction performance on the following 10 minutes. Figure 7 depicts the prediction with the RSF model. The test of the LSTM model on the 10 minute time frame shows an accurate forecast and no miss of a signal peak is detected; the T2G prediction are within the accuracy requirements and can be applied to real-world applications.

V. CONCLUSION

This paper proposes a framework for T2G predictions at an urban intersection to enhance the quality of SPaT messages. The problem was constructed as a time series forecast. The

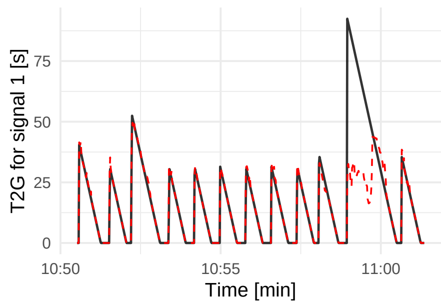

Fig. 5: Prediction sample from the test data set of the T2G for a 10 minutes time frame with the RSF model.

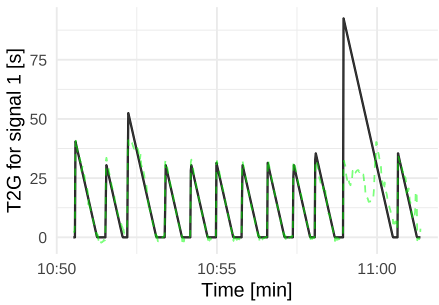

Fig. 4: Prediction sample from the test data set of the T2G for a 10 minutes time frame with the LR model.

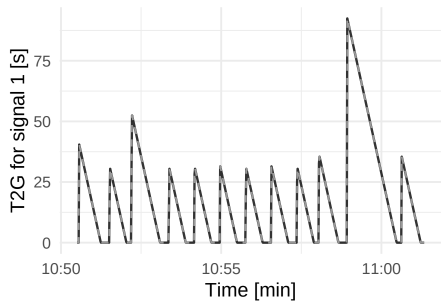

Fig. 6: Prediction sample from the test data set of the T2G for a 10 minutes time frame with the LSTM model.

TABLE I: Model performance comparison for LR, RSF and LSTM on the test data with $\overline{\mathrm{MAE}}$, $\overline{\mathrm{RMSE}}$ and σ_{MAE}, σ_{RMSE}.

	LR				RSF				LSTM			
i	$\overline{\mathrm{MAE}}$	σ_{MAE}	$\overline{\mathrm{RMSE}}$	σ_{RMSE}	$\overline{\mathrm{MAE}}$	σ_{MAE}	$\overline{\mathrm{RMSE}}$	σ_{RMSE}	$\overline{\mathrm{MAE}}$	σ_{MAE}	$\overline{\mathrm{RMSE}}$	σ_{RMSE}
1	2.53	0.73	4.37	1.45	1.24	0.51	3.01	1.29	0.39	0.06	0.41	0.05
2	2.32	0.60	3.63	0.93	1.15	0.41	2.44	0.89	0.37	0.02	0.40	0.02
3	2.30	0.78	3.62	1.31	1.12	0.63	2.44	1.35	0.34	0.03	0.38	0.03
4	2.65	0.78	4.45	1.54	1.15	0.54	2.76	1.48	0.34	0.02	0.39	0.02
5	2.32	0.74	4.05	1.69	1.31	0.59	3.05	1.70	0.33	0.02	0.38	0.02
6	2.54	0.65	4.03	1.00	1.13	0.35	2.41	0.82	0.38	0.02	0.41	0.02
7	2.13	0.60	3.49	0.96	0.64	0.29	1.53	0.89	0.28	0.02	0.35	0.02
8	2.60	0.73	4.45	1.52	1.37	0.57	3.40	1.46	0.25	0.02	0.33	0.01
9	2.17	0.25	3.97	0.39	1.17	0.28	3.50	0.51	0.26	0.03	0.34	0.02
10	2.66	1.09	4.81	1.93	1.24	0.65	3.46	1.68	0.22	0.03	0.31	0.02
11	74.51	11.95	92.96	16.83	69.02	12.05	92.82	16.06	0.42	0.03	0.43	0.02
12	91.11	30.44	111.16	40.28	82.06	30.40	108.10	37.97	0.39	0.06	0.41	0.05

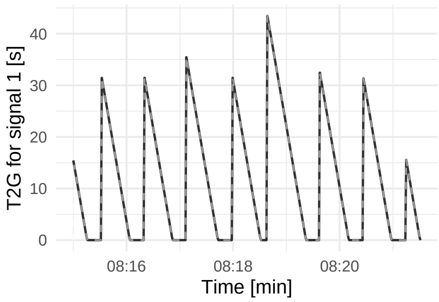

Fig. 7: Prediction of 10 minutes T2G in the peak-hour of another weekday with the LSTM model.

framework implementation is generic and can be applied to any intersection that provides LD and signal data. In the case study, the work was tested on an intersection in the city of Zurich with actuated signal control and public transport priority. Results show that an RSF and LSTM are promising tools for the prediction of residual phase times and they both outperform the baseline model (i.e., LR). Nevertheless, the RSF performance for predictions of public transport traffic lights needs to be further investigated. The LSTM model overcomes this limitation and provides an accurate fit of the time series within the accuracy requirements. Hence, results show that an LSTM should be utilized for T2G predictions in real world applications. Future work will extend the present research with the possibility of predicting the T2G for all signals simultaneously and tests on multiple intersections.

ACKNOWLEDGMENT

The authors would like to thank Felix Denzler from the city of Zurich for collaborating and to provide the data. M. Menendez acknowledges the support by the NYUAD Center for Interacting Urban Networks (CITIES) funded by Tamkeen under the NYUAD Research Institute Award CG001, and by the Swiss Re Institute under the Quantum CitiesTM initiative. L. Ambühl acknowledges the support by ETH Research Grant ETH-27 16-1 under the name of SPEED.

REFERENCES

[1] C. Yu, Y. Feng, H. X. Liu, W. Ma, and X. Yang, "Integrated optimization of traffic signals and vehicle trajectories at isolated urban intersections," *Transportation Research Part B: Methodological*, vol. 112, pp. 89 – 112, 2018.

[2] B. Asadi and A. Vahidi, "Predictive cruise control: Utilizing upcoming traffic signal information for improving fuel economy and reducing trip time," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 3, pp. 707–714, 2011.

[3] S. S. James Misener and S. Dickey, "Investigating the potential benefits of broadcasted signal phase and timing (spat) data under intellidrive sm," *ITS America Annual Meeting*, 2010.

[4] S. A. Fayazi, A. Vahidi, G. Mahler, and A. Winckler, "Traffic signal phase and timing estimation from low-frequency transit bus data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 19–28, 2015.

[5] S. A. Fayazi and A. Vahidi, "Crowdsourcing phase and timing of pre-timed traffic signals in the presence of queues: Algorithms and back-end system architecture," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 3, pp. 870–881, 2016.

[6] C. Wang and S. Jiang, "Traffic signal phases' estimation by floating car data," in *2012 12th International Conference on ITS Telecommunications*, 2012, pp. 568–573.

[7] X. J. Ban, R. Herring, P. Hao, and A. M. Bayen, "Delay pattern estimation for signalized intersections using sampled travel times," *Transportation Research Record*, vol. 2130, no. 1, pp. 109–119, 2009.

[8] Juan Yu and Pei-zhong Lu, "Learning traffic signal phase and timing information from low-sampling rate taxi gps trajectories," *Knowl.-Based Syst.*, vol. 110, pp. 275–292, 2016.

[9] V. Protschky, C. Ruhhammer, and S. Feit, "Learning traffic light parameters with floating car data," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 2015, pp. 2438–2443.

[10] S. Ibrahim, D. Kalathil, R. O. Sanchez, and P. Varaiya, "Estimating phase duration for spat messages," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 7, pp. 2668–2676, 2019.

[11] D. Branston and H. van Zuylen, "The estimation of saturation flow, effective green time and passenger car equivalents at traffic signals by multiple linear regression," *Transportation Research*, vol. 12, no. 1, pp. 47 – 53, 1978.

[12] H. Ishwaran, U. B. Kogalur, X. Chen, and A. J. Minn, "Random survival forests for high-dimensional data," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 4, no. 1, pp. 115–132, 2011.

[13] M. H. D. K. e. a. Nasejje, J.B., "A comparison of the conditional inference survival forest model to random survival forests based on a simulation study as well as on two applications with time-to-event data," *BMC Med Res Methodol*, vol. 17, no. 115, 2017.

[14] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," *Transportation Research Part C: Emerging Technologies*, vol. 54, pp. 187 – 197, 2015.

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