Traffic Speed Prediction with Neural Networks

Umut Can Çakmak, Mehmet Serkan Apaydın and Bülent Çatay

1 Introduction

Road transport has various hazardous and threatening impacts on the environment and human life such as resource consumption, pollution, emission, congestion, and noise. Growing concerns in modern societies about these issues and the quality of life in cities call attention to new methods and approaches in traffic management, transportation planning, and route optimization for both commercial and individual drivers. Many of these methods depend on the estimation of travel time, traffic speed and volume. Recent advancements in Global Positioning Systems (GPS), Geographical Information Systems (GIS), image processing, and sensor technologies enable the real-time collection of these massive data, which can be effectively used to improve the accuracy of the prediction methods.

Early studies mainly collected their data from highway sensors, and GPS data was not common until 2011 [5]. The acquired data usually consists of speed, congestion classification, journey time, and volume. The research on the analysis of the collected data can be categorized as discrete and continuous. Discrete analyses include binary or multiclass classification methods while continuous analyses mainly employ function approximation and time series analysis.

U. C. Çakmak (✉) · B. Çatay
Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, Turkey
e-mail: ucakmak@sabanciuniv.edu

M. S. Apaydın
Department of Computer Science, Istanbul Şehir University, Istanbul, Turkey

U. C. Çakmak · B. Çatay
Smart Mobility and Logistics Lab, Sabanci University, Istanbul, Turkey

N. Kliewer et al. (eds.), *Operations Research Proceedings 2017*, Operations Research Proceedings, https://doi.org/10.1007/978-3-319-89920-6_98

It has been observed that the accuracy of the predictions improve as the number of segments in a route increases [6]. Reference [2] reports around 90% classification accuracy for short-term predictions (up to 5 min) on the highway; however, the results deteriorate in the urban setting. Reference [7] achieves an average of mean absolute deviations (MAD) value of 6.60 km/h for 1-step ahead and 12.47 km/h for 5-step ahead prediction over 20 segments.

In this study, we employ a feedforward neural network (FFNN) to perform a continuous prediction. We are mainly motivated by the work of [7] on irregular data. Our aim is to perform accurate predictions over a relatively longer horizon instead of a fixed point in the future. The remainder of the paper is organized as follows: Sect. 2 introduces the methodology including data collection and cleaning, prediction methods and machine learning concepts. Section 3 presents the experimental setup while Sect. 4 reports and discusses the results. Finally, Sect. 5 concludes with suggestions for future research.

2 Methodology

2.1 Data Collection and Cleaning

The historical speed data is obtained from Başarsoft Information Technologies Inc. It includes floating car speeds collected on Istanbul road network with 1-min time intervals over a 5-month horizon from Oct. 2016 to Feb. 2017.

Since the raw data needed cleaning, we firstly linearly interpolated the missing data and reduced the high speed values to the legal speed limit. Secondly, we used a systematic interpolation technique to smooth out the erratic jumps in observations. For instance, the speed on a particular road segment may change by up to 80 km/h from one minute to the next, which is unrealistic and may be due to data collected from different vehicles en-route or from different road segments nearby. Briefly, our method smoothes the erratic observations by removing the speeds that vary by more than z standard deviations in a given segment, where z is gradually reduced until speed variations are realistic.

2.2 Prediction Methods

In this section, we briefly describe different time-series forecasting methods, where s_t represents the observed speed at time t while f_{t+k} represents the prediction of the speed at time $t+k$.

Naïve Naïve method is the simplest forecasting technique where the prediction is equal to the recently observed speed. This method may perform well for short-term predictions.

$$f_{t+1} = s_t \tag{1}$$

Weighted Moving Average (WMA) The method makes a prediction by taking the weighted moving average of the last n observations as follows:

$$f_{t+1} = w_t s_t + w_{t-1} s_{t-1} + w_{t-2} s_{t-2} + \cdots + w_{t-(n-1)} s_{t-(n-1)} \tag{2}$$

where w_i is the weight associated with the observation at time i with $\sum_{t-(n-1)}^{t} w_i = 1$ and $0 \leq w_i \leq 1$. The benefit of weighted moving average is that it can be tuned to give the most relevant past data more importance [4].

Simple Exponential Smoothing (SES) This method is similar to the weighted moving average where a weight is associated with the most recent observation and another weight is given to the last forecast. This recursive relationship makes the process take into account the whole set of past observations. The formulation is as follows:

$$f_{t+1} = \alpha s_t + (1 - \alpha) f_t \tag{3}$$

where α is the *smoothing constant* and $0 \leq \alpha \leq 1$ [3].

Triple (Winters) Exponential Smoothing This method is developed to handle trend and seasonality simultaneously and it can also be used when the data shows seasonality but no trend. We use this technique because we observe *microseasons* over the course of five months such as the rush hours of weekdays.

$$L_t = \alpha \frac{s_t}{S_{t-M}} + (1 - \alpha)(L_{t-1} + T_{t-1}) \tag{4}$$

$$T_t = \beta(L_t - L_{t-1}) + (1 - \beta) T_{t-1} \tag{5}$$

$$S_t = \gamma \frac{s_t}{L_t} + (1 - \gamma) S_{t-M} \tag{6}$$

$$f_{t+k} = (L_t + k T_t) S_{t+k-M} \tag{7}$$

where L_i is known as *Level* or *Smoothed Observation* at time i and T_i is known as the *Trend* or *Trend Factor* at time i and β is the *trend smoothing constant* which is similar to α and $0 \leq \beta \leq 1$ [3]. Here, S_i is the *Seasonal Index* at time i and γ is the *seasonality smoothing constant* and $0 \leq \gamma \leq 1$ [3]. M is the number of seasons. In our case, the seasons consist of 1-min. time intervals and we have 1440 seasons in a day throughout the entire horizon.

2.3 Machine Learning

Feedforward Neural Networks (FFNN)/Multilayer Perceptrons (MLP) A simple, single layer perceptron has an output unit y_i and input units x_i along with an extra bias unit, a set of weights that connect the inputs and the bias unit to the output [1]. A bias unit is an input unit of $x_0 = 1$. It acts, as can be seen from (8), as the constant in a linear equation.

$$y = \sum_{j=1}^{d} w_j x_j + w_0 x_0 \tag{8}$$

where d is the number of input neurons excluding the bias unit. A multilayer perceptron has the advantage of handling nonlinear functions [1]. The multilayer perceptrons have at least one hidden layer in addition to input and output layers. To train these networks, input and target data are required. In this work, target data is *Cleaned* data, and input data is the forecasts obtained by the methods in Sect. 2.2. Training starts with an initial set of weights and progresses forward over the system to yield an output value. Our network is trained with forward and backpropagation.

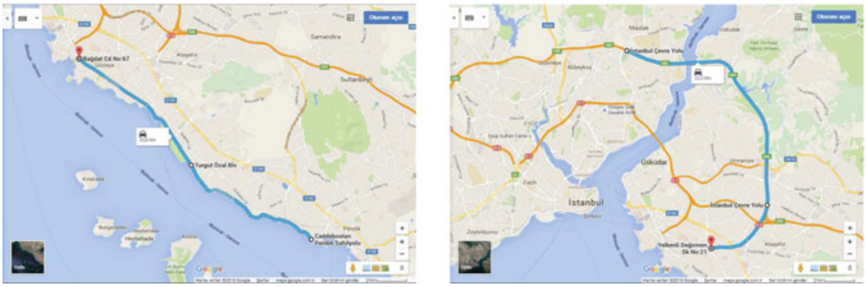
3 Experimental Setup

3.1 Route Selection

We performed our analysis on two different routes in Istanbul (see Fig. 1). The first is an urban route with many intersections that covers 324 segments over a distance of 21.49 km, with mean and median segment lengths of 0.07 and 0.05 km, respectively. The second route is a freeway starting from the European side of the city and crossing the Bosphorus Strait through the FSM Bridge. It covers 63 segments over a distance of 22.75 km, with mean and median segment lengths of 0.36 and 0.25 km, respectively.

3.2 Single Segment Approach (Multi-step Ahead Forecast) (SS-M Network)

Our network (see Fig. 2) has 30 input neurons for each prediction method with 50 hidden neurons in the single hidden layer and 30 output neurons. Each neuron in the input and output layers inputs and outputs a k-step ahead prediction, respectively.



(a) Route1 (b) Route2

Fig. 1 Routes examined

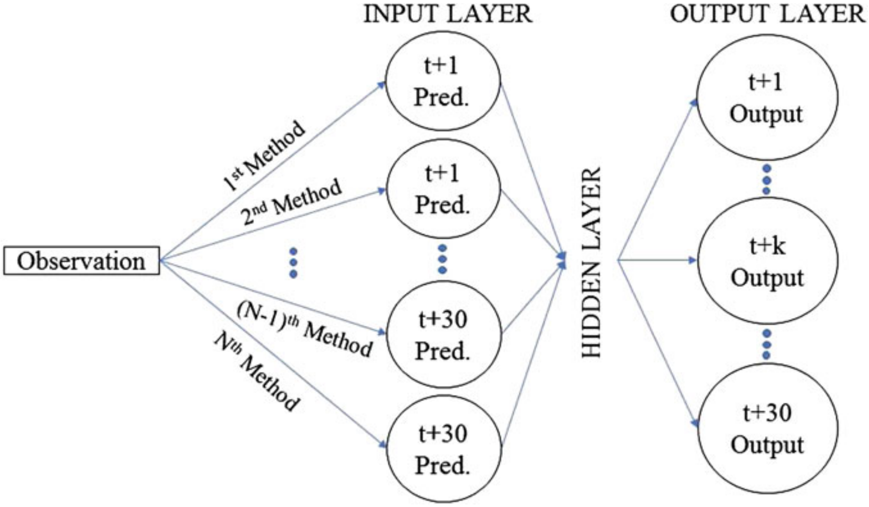


Fig. 2 Single segment approach (multi-step ahead forecast) with N predictive methods (for this work: Naïve, weighted moving average, simple exp. smoothing and winters) (SS-M network)

4 Computational Results

The experiments were carried on a workstation with a 64-bit Windows 7 Professional operating system, a memory of 128 GB, and a 40-core Intel Xeon CPU E5-2640 v4 @ 2.40 GHz processor. We have implemented the FFNN using Keras with Theano and Python 2.7.

We tested NMS (Naïve-Weighted Moving Average-Simple Exponential Smoothing) and NMSW (NMS-Winters) combinations through 30 epochs and a batch size of 1000 with adaptive moment estimation (Adam) optimizer. To prevent overfitting, we also employed a 10% Dropout. We used the following parameters for our prediction methods that are input to the FFNN: Weighted moving average method takes a

Table 1 30-min Test Results by NMS and NMSW predictive methods on SS-M (Proposed Single Segment Multi-step Ahead Forecast Network) and individual predictive methods (Naïve, Weighted Moving Average (WMA), Simple Exponential Smoothing (SES), Winters)

Route	Avg. Seg. Len. (km)	MAD (km/h)						Avg. Train Time (s)	
		Naive	WMA	SES	Winters	NMS	NMSW	NMS	NMSW
1	0.059	0.583	0.595	0.594	0.596	0.470	0.480	738	669
2	0.285	7.867	7.901	7.784	7.749	6.467	6.431	698	647

3-step horizon with three weights: 0.25, 0.50, and 0.25, simple exponential smoothing method takes $\alpha = 0.50$, and Winters method takes $\alpha = 0.45$, $\gamma = 0.20$; thus, only considers seasonality without any trend. In the literature, it is common to assign the parameters intuitively. The first 4.5 months of the dataset were allocated to training while the remaining 15 days were used for testing.

The experimental test results for 30-min prediction horizon are reported in Table 1. Route 1 results are coming from 16 segments spanning 0.95 km while Route 2 results are of 16 segments spanning 4.55 km. In line with [6], we observe that the accuracy of the predictions enhance when the route is split into more segments. This is evident in the fact that the segments of Route 1 return lower error values than those of Route 2. It seems surprising that there is not a significant advantage of employing NMSW over NMS; however, it is worth noting that 30-min-ahead is a relatively short horizon to observe the real effect of seasonality in the prediction.

5 Conclusion

Here we employed FFNN to predict the traffic speed over a 30-min horizon using historical speed data collected in 1-min time intervals. Even though our method requires significant computation effort, its performance is comparable to that of [7], overperforming it on longer term predictions. While their results achieve 12.47 km/h MAD for 5-step ahead prediction over 20 segments, our results for the 16 segments return an average of 0.47–6.43 km/h MAD. To improve our current methods, employing Winters prediction over a longer horizon that reflects seasonal characteristics better than 30-min horizons also seems promising. As further future work, we plan to use recurrent neural networks and also take seasonality into consideration to further improve the prediction accuracy. Random forest regression is also a simple method we can use to combine the individual prediction methods.

Acknowledgements We would like to thank Başarsoft Information Technologies Inc. for providing historical floating car data.

References

1. Alpaydın, E. (2010). *Introduction to Machine Learning*. Cambridge, MA: MIT Press.
2. Khan, R., Landfeldt, B., & Dhamdhere, A. (2012). Predicting travel times in dense and highly varying road traffic networks using STARIMA models. Technical Report.
3. NIST/SEMATECH e-Handbook of Statistical Methods. https://www.itl.nist.gov/div898/handbook/.
4. Stevenson, W. J. (2012). *Operations Management: Theory and Practice*. New York, NY: McGraw-Hill/Irwin.
5. Vlahogianni, E. I., Karlaftis, M. G., & Golias, J. C. (2014). Short-term traffic forecasting: Where we are and where we're going. *Transportation Research Part C: Emerging Technologies*, *43*, 3–19. https://doi.org/10.1016/j.trc.2014.01.005.
6. Wang, J., Mao, Y., Li, J., Xiong, Z., & Wang, W. (2015). Predictability of road traffic and congestion in urban areas. *Plos One*, *10*(4). https://doi.org/10.1371/journal.pone.0121825
7. Ye, Q., Szeto, W. Y., & Wong, S. C. (2012). Short-term traffic speed forecasting based on data recorded at irregular intervals. *IEEE Transactions on Intelligent Transportation Systems*, *13*(4), 1727–1737. https://doi.org/10.1109/tits.2012.2203122.