Transportation Research Part C 124 (2021) 102930

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Predicting cycle-level traffic movements at signalized intersections using machine learning models

Nada Mahmoud *, Mohamed Abdel-Aty, Qing Cai, Jinghui Yuan

Department of Civil, Environmental, and Construction Engineering, University of Central Florida (UCF), Orlando, FL 32816-2450, USA

ARTICLE INFO

Keywords:
Cycle-level
Deep learning
Machine learning
Traffic prediction
Turning movement counts

ABSTRACT

Predicting accurate traffic parameters is fundamental and cost-effective in providing traffic applications with required information. Many studies adopted various parametric and machine learning techniques to predict traffic parameters such as travel time, speed, and traffic volume. Machine learning techniques have achieved promising results in predicting traffic volume. However, the utilized data were mostly aggregated in 5, 10, or 15 min. This study attempts to bridge the research gap by predicting signal cycle-level through and left-turn movements in real-time at signalized intersections. The utilized data were limited to the upstream and downstream intersections at the corridor level. Aiming to achieve this objective, eXtreme Gradient Boosting (XGBoost), Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models were developed using datsets that contain variables from different number of utilized cycles (4, 6, and 8 cycles). The three models were evaluated by calculating Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). The results showed that the performance measures for the three models were close. Meanwhile. the GRU model using variables from six previous cycles outperformed the others. This modelling approach was followed to predict traffic movements for different time horizons (five cycles ahead). The performance measures values were close for the five predicted cycles. It is expected that the model could help in obtaining accurate traffic movement at intersections, which could be used for adjusting adaptive signal timing and improve signal and intersections' efficiency.

1. Introduction

Intelligent Transportation System (ITS) is a multidisciplinary approach that was adopted in many countries to enhance traffic management including transportation planning, signal plans, and traffic safety (Shoup et al., 2013). It enriches users with the information such as traffic congestions levels and travel time estimation. Thus, frequently updated and reliable information that represents future conditions should be collected. In general, the provided traffic data are collected using several techniques such as loop detectors, video cameras, GPS, and various types of sensors. Further, predicting accurate traffic parameters is fundamental and cost effective in providing ITS traffic applications with the required information. In the last decade, extensive efforts were carried out to predict various traffic parameters (e.g, travel time, speed, and traffic volume) using accurate and efficient traffic forecasting models.

Short-term real-time traffic movement counts are considered significant inputs to ITS (TSM&O) traffic management systems.

* Corresponding author.
E-mail addresses: nada.mahmoud@knights.ucf.edu (N. Mahmoud), M.Aty@ucf.edu (M. Abdel-Aty), qingcai@knights.ucf.edu (Q. Cai), jinghuiyuan@knights.ucf.edu (J. Yuan).

https://doi.org/10.1016/j.trc.2020.102930
Received 4 August 2020; Received in revised form 24 November 2020; Accepted 12 December 2020
Available online 5 January 2021

Considering the importance of traffic signals, the predicted traffic counts could be utilized to optimize and manage real-time traffic operations at signalized intersections. Further, traffic prediction could provide real-time information to users, thus, commuters could avoid traffic congestion when forecasted in the upcoming time period (Zhu et al., 2016).

Over the years, many studies developed statistical and machine learning prediction models using temporal information (Lin et al., 2008). However, aiming to improve the efficiency of the developed algorithms spatial data from adjacent intersections were considered along with temporal information (Vlahogianni et al., 2007). In this paper, both spatial and temporal data were utilized to develop machine learning algorithm that aims to predict cycle-level short-term traffic movement counts at signalized intersections.

2. Literature review

In the past decades, several studies have investigated predicting traffic volume at freeways and signalized intersections using parametric and non-parametric models. They focused on predicting the traffic volume in the next 5-min and 15-min (Qiao et al., 2013). For instance, Autoregressive Integrated Moving Average (ARIMA) model and its derivatives were considered regular practices in predicting short-term traffic parameters using time series data. Utilizing ARIMA model to predict traffic parameters was first introduced by Ahmed and Cook in 1979 (Ahmed and Cook, 1979). This approach was found to be helpful in many traffic predictions (Kamarianakis and Prastacos, 2005; Stathopoulos and Karlaftis, 2003; Williams and Hoel, 2003). Furthermore, in the literature, diverse variants of the ARIMA model such as Seasonal Autoregressive Integrated Moving Average (SARIMA), Space–Time Autoregressive Integrated Moving Average (STARIMA), and Generalized Space-Time Autoregressive Integrated Moving Average (GSTARIMA) models were proposed to predict time series short-term traffic volumes. The results showed that the performance measures of the developed prediction models were acceptable specially at the entry segments of the study area. However, developing GSTARIMA model requires more historical data to predict traffic parameters (Ghosh et al., 2007, 2005; Guo et al., 2013; Kamarianakis and Prastacos, 2005; Min et al., 2010, 2009). Other parametric models have been developed to predict traffic parametrs such as traffic volume (Wu et al., 2016; Zhan et al., 2018), travel speed, and travel time (Rong et al., 2015). State-space models such as Kalman Filter (KF) and Wavelet Kalman Filter (WKF) have been utilized to predict travel time and traffic flow at interstate highways. The prediction results showed that the WKF model outperformed the direct KF when predicting traffic flow (Chen and Chien, 2001; Xie et al., 2007). In another study, the Adaptive Kalman Filter (AKF) model was developed to predict 15-min traffic volume at highways. The model was found to be adaptable when the traffic flow is unsteady (Guo et al., 2014).

Some studies developed algorithms that consider both spatial and temporal (i.e., spatiotemporal) features of traffic flow. For instance, Spatial-Temporal Random Effect (STRE) model was presented by Wu et al. (2016) to estimate traffic volumes. The model was able to estimate traffic volume effectively with Mean Absolute Percentage Error (MAPE) of 16% (Wu et al., 2016). Moreover, Bayesian Network (BN) models were widely applied for traffic flow forecasting when considering spatiotemporal features (Castillo et al., 2008). Further, Linear Conditional Gaussian Bayesian Network (LCG-BN) model was utilized by Zhu et al. (2016) for short-term traffic counts predictions. The performance of the developed model showed that the prediction accuracy increased significantly when utilizing speed data along with spatial data. However, the trade off between model's complexity and prediction accuracy makes it difficult to determine which variables to be included in the model (Li et al., 2019).

One the other hand, the K-Nearest Neighbors (KNN) is a non-parametric model that has been widely developed to predict short-term traffic volume taking into consideration its spatial features. The first KNN model was developed by Davis and Nihan in 1991. They developed the model using freeway data and compared it to a simple univariate linear time-series method. The KNN method performed comparably to, but not better than, the linear time-series approach (Davis and Nihan, 1991). Further, other KNN techniques were also proposed in recent studies to predict short-term traffic volume at signalized intersections such as K-Nearest Neighbor–Trend Adjustment Model (KNN-T), K-Nearest Neighbor Non-Parametric Regression (KNN-NPR), K-Nearest Neighbor enhanced by Least Square Probabilistic Classification (KNN-LSPC), and Spatial-Temporal Weighted K-Nearest Neighbor (STW-KNN) models. In most cases, KNN techniques were found to have significantly higher efficiency and scalability than parametric models (i.e., ARIMA and Kalman Filter) due to higher flexibility and extendibility of the KNN techniques (Clark, 2003; Guo et al., 2018; Qiao et al., 2013; Xia et al., 2016; Yoon and Chang, 2014; Zheng and Su, 2014).

Aiming to promote the accuracy and efficiency of short-term traffic flow prediction, many studies adopted different machine learning techniques such as Neural Networks (NN) and Gradient Boosting Decision Trees (GBDT). A basic three-layered back-propagation Multilayer Perceptron (MLP) model was utilized to develop the first NN by Clark et al. (1993). In recent years, many studies adopted Neural Networks to predict short-term traffic volume at signalized intersections. Vlahogianni et al. (2005) concentrated on developing a neural network model for forecasting short-term traffic volume at signalized intersections on congested urban arterials. The model was developed based on error back-propagation feed-forward MLP techniques and genetic algorithms by considering both spatial and temporal traffic flow features. In order to test the model, univariate and multivariate data from signalized urban arterials were used to evaluate the developed neural network. The results showed that the multivariate data were helpful in the case of multiple-step forecasting. Moreover, the results showed that neural networks model performed better than the univariate ARIMA and multivariate State-Space models in forecasting short-term traffic flow (Vlahogianni et al., 2005). In another study, in order to predict traffic volume, a model that combines both ARIMA and NN models was developed and compared with basic ARIMA and NN models ANN models' performance in forecasting traffic flow. The results showed that the combined ARIMA and NN model produced more accurate estimated traffic flows than using a single model (ARIMA or NN) (Chang et al., 2000). Other NN techniques (i.e., recurrent Wavelet Neural Network (WNN), Radial Basis Function Neural Network (RBFNN), Bayesian Combined Nerual Network (BCNN), and Time Lag Recurrent Network (TLRN)) were developed using data aggregated in 5, 10, or 15 minutes interval in order to carry out an effective short-term traffic flow prediction. The results showed that the NN techniques outperformed the ARIMA techniques in many cases.

Further, the BCNN model outperformed the single neural network models (NN and RBFNN) in predicting traffic flow at freeways (Dia, 2001; Jiang et al., 2005; Zheng et al., 2006; Zhu et al., 2014). Other studies utilized a machine learning approach by developing GBDT model which takes into consideration the relation between the search sliding time and feature extension. The studies showed that the GBDT model's predictions for different time horizons (5, 10, and 15 minutes ahead) were more accurate than both Support Vector Machine (SVM) and Back Propagation Neural Networks (BPNN) models (Cheng et al., 2019; Liu et al., 2017; Xia and Chen, 2017; Yang et al., 2017; Zhang et al., 2016; Zhang and Haghani, 2015; Zhao et al., 2017).

Recently, deep learning techniques were found to have promising results in many traffic predictions' research due to their flexibility and the ability to model nonlinearity and capture spatiotemporal features in traffic flow. For instance, Long Short-Term Memory (LSTM) was widely developed in recent researches to predict short-term traffic parameters (e.g., traffic volume, travel time, traffic speed and queue length) by learning from the time series data (Cui et al., 2018; Rahman and Hasan, 2020). Further, other LSTM techniques such as Singular Point Probability LSTM (SPP-LSTM) and Convolutional Neural Network LSTM (CNN-LSTM) were developed to predict more accurate real-time traffic volume at signalized intersections (Li and Ban, 2019; Liu et al., 2017; Yang et al., 2019; Zhao et al., 2017; Zheng et al., 2019). The results showed that these models outperformed baseline models in terms of accuracy (Zhang et al., 2019). The Gated Recurrent Unit (GRU) model was also utilized to achieve better prediction results. It adopts the same gated mechanism as LSTM models. Further, it has simple structure as well as faster training abilility. The GRU model was found to have better results than the History Average (HA) model, the ARIMA model, and the Support Vector Regression (SVR) model (Cao et al., 2017; Fu et al., 2016; Zhao et al., 2019).

In the literature, few studies adopted predicting/estimating turning movement counts at signalized intersections. For instance, Path

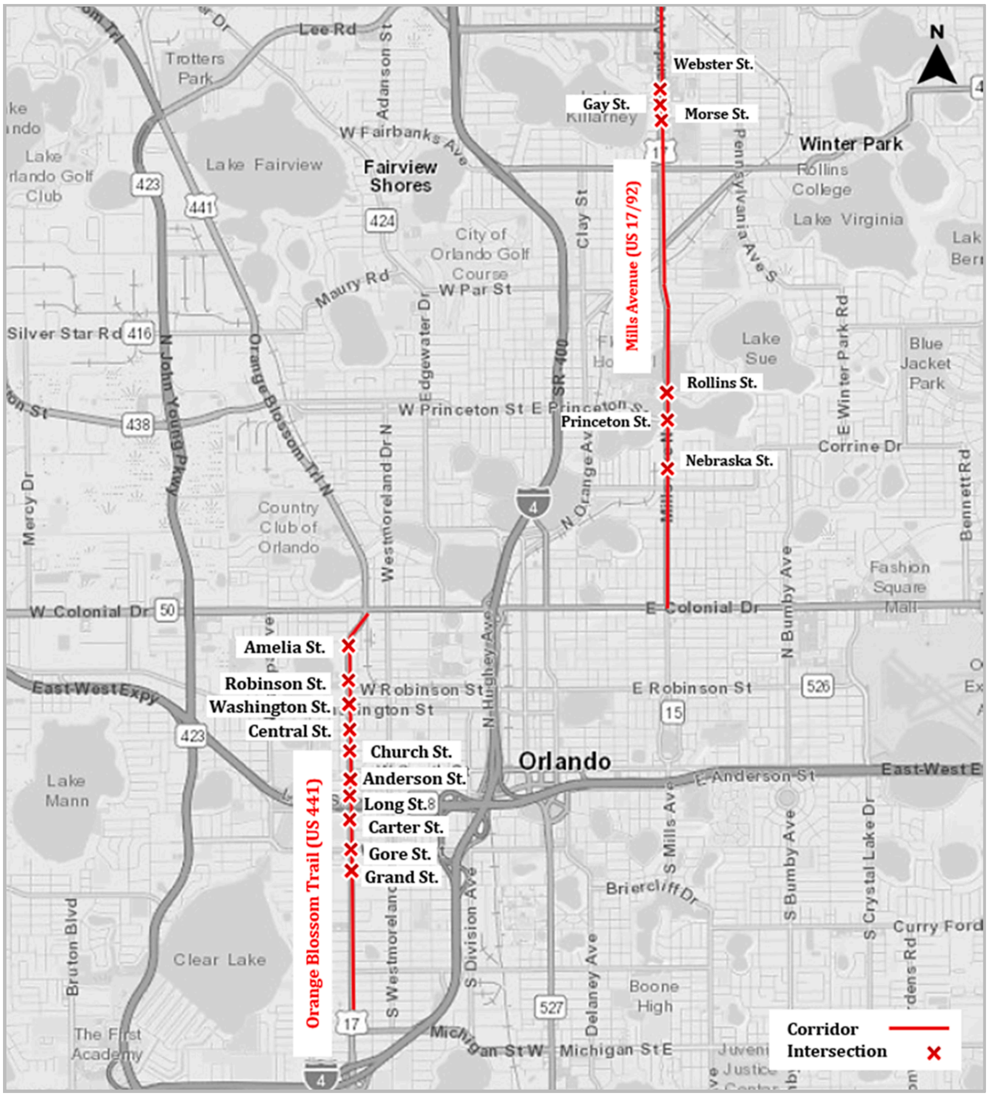

Fig. 1. Study area.

Flow Estimator (PFE) model was utilized to estimate turning movement counts based on the total inbound and outbound peak hour volumes. The PFE model achieved acceptable results at main corridor. However, the minor corridors estimation wasn't good (Chen et al., 2012). In another study, hourly turning movement counts were estimated based on the approach volumes. The researchers developed a Neural Network (NN) model that was trained to analyze the relation between approach volumes and turning movements. The estimation results were acceptable but limited to peak hour period (Ghanim and Shaaban, 2018). Further, Partial Least Square (PLS) model was developed in a recent study to predict 15-min short-term turning movements at signalized intersections using trajectory data. The results showed that the PLS model outperformed other prediction models (i.e., ARIMA, KNN and SVR) (Li et al., 2020).

In summary, previous research focused on predicting traffic parameters such as traffic volume, travel speed, and travel time in 5, 10, and 15 min (Qiao et al., 2013). Moreover, few studies aimed to predict turning movement counts at signalized intersections. To the best of the authors' knowledge, no study attempted to predict traffic movement counts at the cycle level. Moreover, no study adopted turning movements forecasting for the next five cycles. Therefore, this study aims to bridge the gap by predicting cycle-level traffic movement counts at signalized intersections. The main contributions of this paper are as follow:

(1) Developing a machine learning model that is able to predict accurate short-term cycle-level through and left-turn movements at signalized intersections at the corridor level.
(2) Training the model using cycle-level aggregated traffic volume data from upstream and downstream intersections as well as signal data at the corridor level (only North-South direction). No data from upstream or downstream intersections in East-West direction were available and thus utilized.
(3) Exploring the model accuracy in different time horizons (up to five cycles ahead).
(4) The predicted cycles could be utilized in several applications to improve the corridor's traffic situations such as predicting traffic queues and improve traffic management during incidents.

In general, this paper aims to develop a machine learning model that could predict precise and accurate through and left-turn movements at signalized intersections at cycle level that outperformed other proposed models in literature. Moreover, it aims to predict different time horizons with good performance metrics. Aiming to achieve these objectives and based on the previous literature results; three machine learning models were adopted: LSTM, XGBoost, and GRU. This is the first attempt to apply XGBoost and GRU models to predict turning movements at signalized intersections.

2.1. Data collection and proccessing

In this research, one-month real-time traffic data were collected from 16 intersections along two main corridors (US 441 and US 17/92) in Orange County, Florida. The data were obtained from the GRIDSMART system using provided IP addresses from Florida Department of Transportation (FDOT). The GRIDSMART system is a detection system that provides both turning movement counts and traffic signal data based on automated analysis for the videos that were collected using single-camera system. Some studies utilized both signal data and traffic volume data in their analysis to improve the prediction results (Kong et al., 2013; Zheng and Liu, 2017). Fig. 1 shows the study area and the intersections that were utilized in developing the machine learning prediction model. The weekday data were utilized in this research.

The 16 intersections were divided into 6 groups, each three consecutive intersections create an individual group. The middle intersection is considered the target one at which through and left-turn movements were predicted. The intersections in each group were defined as upstream intersection (intersection 1), target intersection (intersection 2), and downstream intersection (intersection 3). An intersection could take different position when included in multiple groups. Table 1 lists the different groups of intersections in the study area.

The GRIDSMART detection system provides raw data that represent each single vehicle as a record. These traffic counts data were provided for each lane seperately. Further, GRIDSMART also provides single event data for any change in signal phasing at intersections ("GRIDSMART USERGUIDE," 2019). Both traffic counts and signal data were utilized in developing the short-term prediction model. The data were aggregated at the cycle level based on the provided signal data. First, signal data were processed to calculate the start of cycle, end of cycle, and cycle length for each intersection. The start and end of cycles were determined based on the coordination phase at the intersection. They were represented by the start of red and the end of yellow for the intersection, respectively. Also, the signal data were utilized to calculate each movement's green time per cycle. On the other hand, the traffic

Table 1
Intersections Groups.

Corridor	Group Number	Intersection 1	Intersection 2 (Target intersection)	Intersection 3
Mills Corridor (US 17/92)	Group 1	Rollins St.	Princeton St.	Nebraska St.
	Group 2	Webster St.	Gay St.	Morse St.
OBT Corridor (US 441)	Group 1	Amelia St.	Robinson St.	Washington St.
	Group 2	Washington St.	Central St.	Church St.
	Group 3	Anderson St.	Long St.	Carter St.
	Group 4	Carter St.	Gore St.	Grand St.

counts data for all intersections were combined into one dataset. Clockwise labelling was given to each approach based on the travel direction; aiming to generalize the approach movement prediction. For example, number "1" was given to the approach of interest at which through and left-turn movements will be predicted. Then, the other approaches were renamed "2", "3", and "4" respectively in clockwise direction. Both signal data and traffic counts were combined. Finally, all the features were aggregated at the cycle-level. This aggregation was carried out based on the cycle data for the middle intersection (target intersection). The final dataset contains 53 features. Table 2 shows the processed features and their description.

In this research, different number of time slices (4, 6, and 8 cycles) were utilized to develop the prediction model. Hence, different datasets were compiled to be utilized in the model development. The number of features in each dataset is a function of N as shown in Equation (1). Where N is the number of utilized time slices.

$$Number\ of\ features = 53^{*}N \tag{1}$$

Table 2
Features Labelling and Description.

Variable	Description
1Ent	Total volume entering intersection (direction of movement)
1LD	Left-turn volume at downstream intersection (direction of movement)
1LGD	Left-turn green time at downstream intersection (direction of movement)
1LGM	Left-turn green time at middle intersection (direction of movement)
1LGU	Left-turn green time at upstream intersection (direction of movement)
1LU	Left-turn volume at upstream intersection (direction of movement)
1RD	Right turn volume at downstream intersection (direction of movement)
1RU	Right turn volume at upstream intersection (direction of movement)
1SD	Through movement volume at downstream intersection (direction of movement)
1SGD	Through movement green time at downstream intersection (direction of movement)
1SGM	Through movement green time at middle intersection (direction of movement)
1SGU	Through movement green time at upstream intersection (direction of movement)
1SU	Through movement volume at upstream intersection (direction of movement)
2LD	Left-turn volume at downstream intersection (East-West direction)
2LGD	Left-turn green time at downstream intersection (East-West direction)
2LGM	Left-turn green time at middle intersection (East-West direction)
2LGU	Left-turn green time at upstream intersection (East-West direction)
2LU	Left-turn volume at upstream intersection (East-West direction)
2RD	Right turn volume at downstream intersection (East-West direction)
2RU	Right turn volume at upstream intersection (East-West direction)
2SD	Through movement volume at downstream intersection (East-West direction)
2SGD	Through movement green time at downstream intersection (East-West direction)
2SGM	Through movement green time at middle intersection (East-West direction)
2SGU	Through movement green time at upstream intersection (East-West direction)
2SU	Through movement volume at upstream intersection (East-West direction)
3Ent	Total volume entering intersection (opposed to direction of movement)
3LD	Left-turn volume at downstream intersection (opposed to direction of movement)
3LGD	Left-turn green time at downstream intersection (opposed to direction of movement)
3LGM	Left-turn green time at middle intersection (opposed to direction of movement)
3LGU	Left-turn green time at upstream intersection (opposed to direction of movement)
3LU	Left-turn volume at upstream intersection (opposed to direction of movement)
3RD	Right turn volume at downstream intersection (opposed to direction of movement)
3RU	Right turn volume at upstream intersection (opposed to direction of movement)
3SD	Through movemen volume at downstream intersection (opposed to direction of movement)
3SGD	Through movement green time at downstream intersection (opposed to direction of movement)
3SGM	Through movement green time at middle intersection (opposed to direction of movement)
3SGU	Through movement green time at upstream intersection (opposed to direction of movement)
3SU	Through movement volume at upstream intersection (opposed to direction of movement)
4LD	Left-turn volume at downstream intersection (West-East direction)
4LGD	Left-turn green time at downstream intersection (West-East direction)
4LGM	Left-turn green time at middle intersection (West-East direction)
4LGU	Left-turn green time at upstream intersection (West-East direction)
4LU	Left-turn volume at upstream intersection (West-East direction)
4RD	Right turn volume at downstream intersection (West-East direction)
4RU	Right turn volume at upstream intersection (West-East direction)
4SD	Through movement volume at downstream intersection (West-East direction)
4SGD	Through movement green time at downstream intersection (West-East direction)
4SGM	Through movement green time at middle intersection (West-East direction)
4SGU	Through movement green time at upstream intersection (West-East direction)
4SU	Through movement volume at upstream intersection (West-East direction)
CycleLengthD	Cycle length at downstream intersection
CycleLengthM	Cycle length at middle intersection
CycleLengthU	Cycle length at upstream intersection

Note: Previous cycle variables labels starts with "No. of previous cycle" + letter "P" as indication of the previous cycle, then the same label as current cycle variables

3. Methodology

Based on the literature review and preliminary investigation, machine learning models have shown good potential and promising performance when utilized in predicting short-term traffic parameters (i.e., speed, travel time, and traffic volume). Hence, in this research, one machine learning model (eXtreme Gradient Boosting (XGBoost)) and two deep learning models (Long-Short Term Memory (LSTM) and Gated Recurrent Unit (GRU)) were proposed to predict cycle-level through and left-turn movements at signalized intersections using previous cycles' data. Fig. 2 shows the adopted methodology in this research.

First, the data were processed, and different datasets were compiled based on the number of utilized cycles in prediction (4, 6, or 8 cycles). Then the data were split into train and test datasets with a ratio of 4:1. Both dependent and independent variables were normalized using standard scaler. The models were trained, tuned, and tested to predict through and left-turn movements at the corridor level (North-South Direction). Right turn movements weren't considered in this research since they are few per cycle as well as they are not controlled by the signal timing. Hence, the concentration was on through and left-turn movements' predictions. It should be noted that the models were developed to predict one traffic movement at a time. Further, the hyperparameters in each developed model (XGBoost, LSTM, or GRU) were tuned for each utilized dataset (4, 6, or 8 cycles data). Thus, values of the hyperparameters were different in the developed models. Eventually, the prediction results were compared to the GRIDSMART observations. Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) were utilized to capture the prediction performance of the developed models. Equations (2) and (3) defines the performance measures (MAE and RMSE), respectively (Zhao et al., 2019).

$$MAE = \frac{1}{n}\sum\left|y_{obs} - y_{pred}\right| \tag{2}$$

$$RMSE = \sqrt[2]{\frac{1}{N}\sum\left|y_{obs} - y_{pred}\right|^2} \tag{3}$$

where N is the number of observations, y_{obs} is the observed traffic movement from GRIDSMART system, and y_{pred} is the predicted traffic movement from the trained models. Afterwards, the developed models were compared and the best modelling approach as well as the best number of utilized previous cycles in the dataset were chosen based on the calculated performance measures. Finally, the selected model was utilized to predict through and left-turn movements for different future time horizons (up to five cycles).

3.1. Extreme gradient boosting model

The eXtreme Gradient Boosting (XGBoost) is a scalable and efficient machine learning algorithm that adopt the Gradient Boosting Decision Trees (GBDT) framework (Friedman et al., 2000). It improves learning accuracy by generating weak learners at each step. In

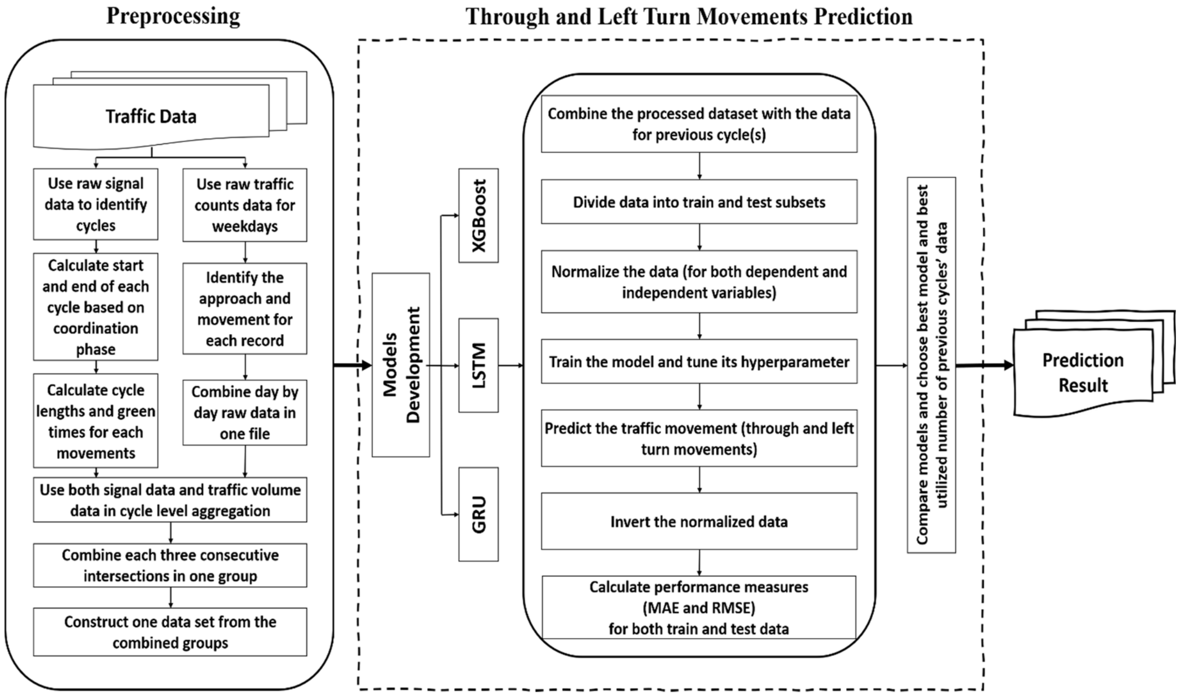

Fig. 2. Research methodology.

each step, these weak learners correct the model error by minimizing the loss function and adjusting the weights. Eventually, the developed weak learners accumulated to generate a high accurate decision tree model (Pan, 2018; Zhang and Haghani, 2015). XGBoost is considered more accurate than the regular Gradient Boosting model as it's able to find the best tree model by utilizing more accurate approximation. It provides more information about the gradients direction by utilizing the second order derivative of the loss function as approximation. Moreover, XGBoost model has advanced regularization. The regression tree model can be illustrated in Equation (4) (Dong et al., 2018).

$$f = w_{q(x(t))} \tag{4}$$

where q represents each regression tree structure that maps a sample to the corresponding leaf index, w represents the leaf weight, and $x(t)$ is the train data sample at iteration t. For the XGBoost model, the regularized loss function at iteration t can be written as shown in Equation (5) (Chen and Guestrin, 2016).

$$\mathscr{L}^{(t)} = \Sigma_{i=1}^{n} l\left(y_i, \widehat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega\left(f_t\right) \tag{5}$$

where l is a differential convex loss function, y_i is the target and $\widehat{y}_i^{(t)}$ is the prediction of the i-th instance in the t-th iteration, x_i is the i-th instance of the training sample, and $\Omega(f_t)$ is illustrated in Equation (6) (Dong et al., 2018).

$$\Omega(f_t) = \gamma T + \frac{1}{2}\lambda \Sigma_{i=1}^{T} w_i^2 \tag{6}$$

where T is the leaf nodes number, γ and λ are the regularization parameters. In this research, the XGBoost algorithm was utilized to predict cycle-level through and left-turn movements at signalized intersections. Variables reduction was carried out based on the features importance values which is computed based on the Mean Squared Error (MSE) for each split in each tree. Hence, only variables with significant influence were included in the developed XGBoost model. Aiming to achieve high prediction accuracy, the models' hyperparameters were tuned (i.e., shrinkage, interaction depth, and number of utilized trees). Furthermore, to avoid overfitting, cross-validation was considered and the number of folds were tuned (Greenwell et al., 2019).

3.2. Long-short term memory model

Long-Short Term Memory (LSTM) is an effective deep learning model that deals effectively with sequential data problems. It was first proposed in 1997 to solve the gradient vanishing or blowing up problems in a Recurrent Neural Network (RNN) (Hochreiter and Schmidhuber, 1997). LSTM has a gated sophisticated structure that is similar to the RNN. Its architecture includes the hidden layer component which is not in the RNN architecture. The gated structure allows the LSTM network to keep the significant information. Further, the forget units can determine the optimal time lags by determining which information needs to be forgotten (Fu et al., 2016). The backpropagation algorithm adjusts the weight matrices automatically and allows the model to control the influence of each input parameter. Fig. 3 shows the typical structure of LSTM cell. A standard LSTM structure includes an input gate (i_t), an output gate (o_t), a forget gate (f_t), a memory cell (C_t), and a hidden state (h_t). Equations (7) to (13) illustrates the calculations of the LSTM components as well as the output vector (Y_T).

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \tag{7}$$

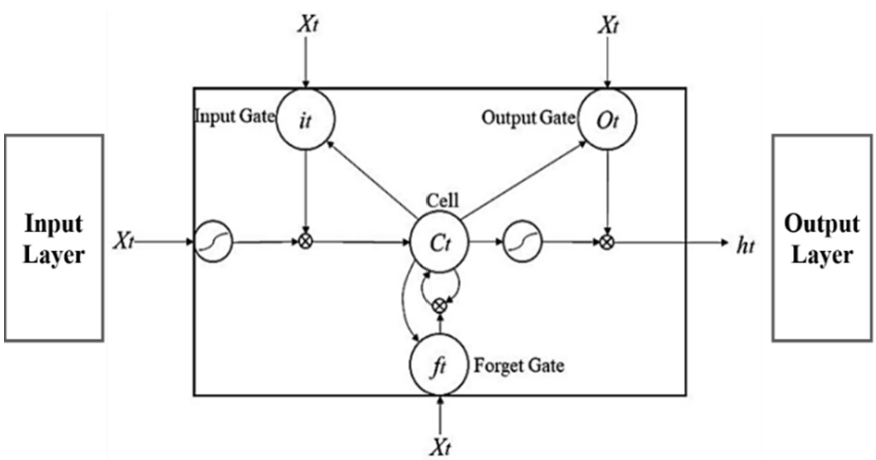

Fig. 3. Typical LSTM cell tructure (Graves et al., 2013).

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \tag{8}$$

$$f_t = \sigma_g\left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{9}$$

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C}_t \tag{10}$$

$$\widetilde{C}_t = tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{11}$$

$$h_t = o_t * \tanh(C_t) \tag{12}$$

$$Y_t = W_{yh} h_{t-1} + b_y \tag{13}$$

where W_i, W_o, W_f represents the weight matrices from the hidden layer input to the three gates. While U_i, U_o, U_f represents the weight matrices from the output of the previous cell and the three gates. $W_c$is the weight metrices from the input of the hidden layer and the input cell state. U_c is the weight metrices from the input of the output of the previous cell and the input cell state. The b_i, b_o, b_f, $b_c$represents the bias vectors. σ_g is the sigmoid activation function, while t is the time iteration. In this research, the LSTM model was trained and validated using the processed data for different time steps (4, 6, and 8 cycles).

The network architecture of the developed LSTM model consists of one input layer, four LSTM hidden layers, and one output layer. The Rectified Linear Unit (ReLU) was used as an activation function. Moreover, aiming to prevent overfitting; dropout layers were included. The input layer shape was (S, T, F) where S is the number of observations, T is the time-steps, and F is the number of features. The processed time-series data were utilized in developing LSTM models for different time-slices (4, 6, and 8 cycles). Hence, the input layer shape varied based on the time-slice value. The output layer represents the predicted number of through or left-turn movements from the developed models. On the other hand, finding the ideal values of the model's hyperparameters is vital as they could significantly influence the resulting accuracy. Hence, to achieve the best performance; automatic hyperparameters tuning was carried out. The grid search approach was followed to explore the alternative configurations of the developed model (Koch et al., 2017). In this research, the grid included the following hyperparameters: batch size, training epochs, learning rate, and optimizer.

3.3. Gated recurrent unit model

The Gated Recurrent Unit (GRU) has a gated structure that is similar to the RNN architecture as in LSTM. However, GRU considered simpler to compute and implement than the LSTM model. It was first proposed in 2014 by Cho et al (Cho et al., 2014). The typical structure of the GRU cell includes a reset gate (which is similar to LSTM's forget gate) and an update gate (Fu et al., 2016). The update gate (z_t)determines whether to update the hidden state (h) or not. Further, the reset gate (r_t) determines whether to ignore the previous hidden state or not. Where $[]_t$ denotes the t-th element of a vector. The calculations for GRU components and the activation function are illustrated in Equations (14) to (17) (Cho et al., 2014).

$$r_t = \sigma(W_r x_t + U_r h_{t-1}) \tag{14}$$

$$z_t = \sigma(W_z x_t + U_z h_{t-1}) \tag{15}$$

$$h_t = z_t * h_{t-1} + (1 - z_t)\widetilde{h}_t \tag{16}$$

$$\widetilde{h}_t = \phi(W x_t + U(r \odot h_{t-1})) \tag{17}$$

where x is the input, t is the time iteration, $\sigma$is the sigmoid activation function, and $(r \odot h_{t-1})$ denotes elementwise (Hadamard) multiplication between the reset gate (r) and the previous hidden state (h_{t-1}). While $W_r, W_z, U_r, U_z$are the weight matrices. Equation (17) indicates that when the reset gate approaches zero, the hidden state reset and drop any insignificant information. On the other hand, in order that the RNN remembers long-term information, the update gate determines the amount of information that will transferred between consecutive hidden states (similar to LSTM's memory cell) (Bengio et al., 2013). The GRU model's network structure is similar to the LSTM model. They both have the same input layer and output layer. Dropout layers were also included to prevent overfitting. Further, the same grid search approach was followed to tune the model's hyperparameters automatically (batch size, training epochs, learning rate, and optimizer).

4. Results and discussion

Initially, simple GRU models were developed to predict through and left-turn movements using only the duration of green time of the predicted movement as the independent variable. The models were developed using 6 cycles' data as an input. The preliminary results show that the calculated performance measures were significantly lower in the final model. This indicates that other variables have significant influence on predicting turning movements besides the green time and must be included in the developed models in order to improve the prediction accuracy. Hence, three datasets were processed to be utilized in developing the final prediction models, one for each utilized time slices (4, 6, and 8 cycles). The number of variables included in the model for these datasets were 212, 318, and 424, respectively. Aiming to achieve the research objective, a total of eighteen models were developed to predict cycle-level through and left-turn movements at signalized intersections. Afterwards, the developed models' performance measures (MAE and RMSE) were calculated, compared, and the best modelling approach was determined. Table 3 shows the calculated performance measures for the developed models and the utilized number of cycles in each model to predict the next cycle.

The performance measures values for both machine learning approach (XGBoost) and deep learning approach (LSTM and GRU) didn't differ much. However, the GRU model that was developed using the previous six cycles data achieved a little better performance than the LSTM and XGBoost models. Its performance metrics were the best in predicting both through and left-turn movements. The MAE and RMSE values for test data were 5.26 and 9.80 for through movements and 0.86 and 1.64 for left-turn movements, respectively. Moreover, the results concluded that utilizing six prior cycles achieved the best performance and no improvement occurred when increasing the utilized cycles to eight. Figs. 4 and 5 show plots for samples of the predicted through and left-turn traffic movements, respectively, and their corresponding observed volumes. The plots show that the predicted through and left-turn movements follow the same patterns of the corresponding observed movements. Moreover, the figures indicate the potential of the model to capture traffic variability and fluctuation in consecutive cycles. For instance, the model could predict a through movement of 80 vehicles then a movement of 40 vehicles in two consecutive cycles. This could be very useful to improve traffic management during incidents and optimize signal control systems.

As the performance of the GRU model was better than both LSTM and XGBoost models; the same approach was followed to predict through and left-turn movements for the next time horizons (5 cycles ahead). Each model was developed to predict a single forecast horizon (one cycle in the furture). Moreover, the dataset that contains 6 previous cycles were utilized in developing these models. Fig. 6 shows the architecture of the followed approach to predict five cycles ahead.

Similar performance measures were calculated to evaluate the developed models as shown in Table 4. The results showed a slight increase in both MAE and RMSE when the number of predicted cycle increases. Nevertheless, the performance measures for different time horizons were on the same range. For instance, MAE and RMSE for the fifth cycle's through movement prediction model were increased by 11.9% and 2%, respectively, when compared to the base case (predicting one cycle ahead). Further, for left-turn movements prediction, the values increased by 3.4% and 4.21%, respectively. Hence, the GRU model was able to predict traffic movements for five cycles in the future with acceptable MAE and RMSE values. This could be benefitial in many applications to improve corridors' traffic management during congenstion and incidents, adjust adaptive signals, and provide better TSM&O strategies.

In order to check the transferability of the developed models; GRU models were trained on one corridor (US 441) and validated on the other one (US 17/92). The models were trained to predict the next cycle using previous six cycles' data. The calculated performance measures were promising. They showed that the results were close to the corresponding developed models (Test data performance for through movements: MAE = 5.96, RMSE = 9.6; for left turn movements: MAE = 2, RMSE = 3.25). Training data limitations presents one of the major obstacles to potential transferability of machine learning models. It's not optimal to transfer a model trained on only four intersections data to predict two intersections' turning movements. Hence, more work with larger sample of data could be considered in future work to capture all the variabilities in intersections data.

Table 3
Developed Models Performance Measures.

Developed Model	No. of Cycles	Train Data MAE	Train Data RMSE	Test Data MAE	Test Data RMSE
Through Movement					
LSTM	4	5.24	9.43	5.80	10.41
	6	4.72	9.54	5.76	10.47
	8	5.10	9.07	5.54	9.82
XGBoost	4	4.83	8.55	5.50	9.84
	6	4.64	8.33	5.41	9.86
	8	4.56	8.30	5.37	9.85
GRU	4	4.88	8.89	5.38	9.87
	6	**4.26**	**8.11**	**5.26**	**9.80**
	8	4.24	8.12	5.53	9.95
Left-turn Movement					
LSTM	4	0.82	1.62	0.90	1.73
	6	0.97	1.77	0.99	1.82
	8	0.81	1.58	0.87	1.68
XGBoost	4	0.79	1.50	0.94	1.71
	6	0.77	1.23	0.93	1.67
	8	0.76	1.20	0.93	1.65
GRU	4	0.82	1.58	0.87	1.66
	6	**0.83**	**1.60**	**0.86**	**1.64**
	8	0.84	1.62	0.86	1.66

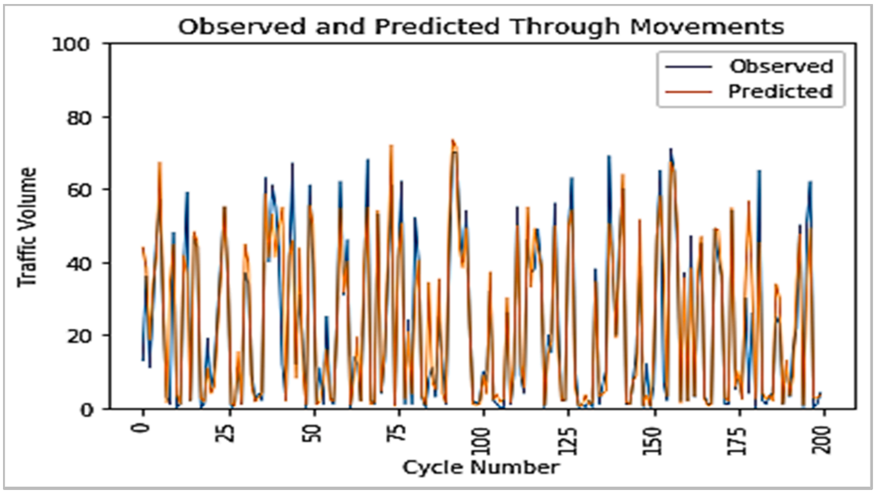

Fig. 4. Sample of observed and predicted through movements.

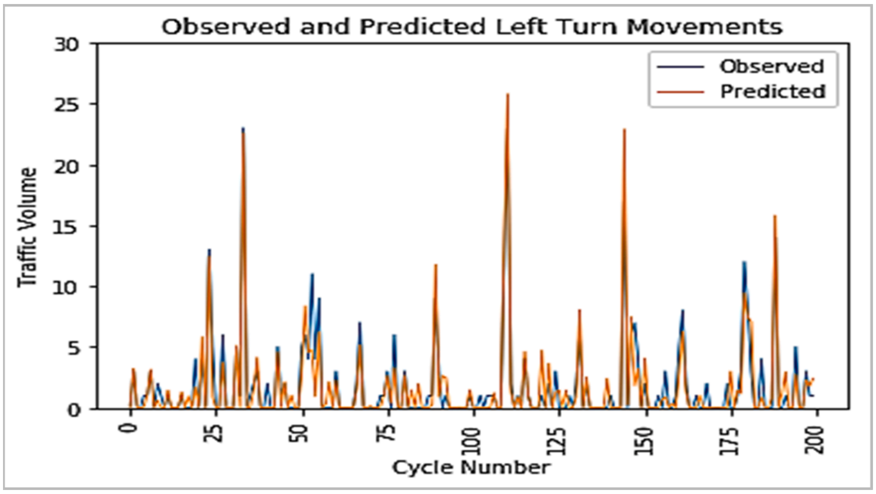

Fig. 5. Sample of observed and predicted left-turn movements.

5. Conclusions

In this paper, XGBoost, LSTM and GRU models were developed and compared to predict short-term through and left-turn movements at signalized intersections at corridor level. Hence, real-time data from 16 intersections along two main corridors were collected. The data included both traffic movement counts and signal data for each real time event. Afterwards, signal data were compiled with traffic movement counts and processed to aggregate traffic counts and calculate green times in cycle-level. Three datasets were created to be utilized in models' development. Each dataset contains different number of previous cycles (4, 6, and 8 cycles) and aims to predict the next cycle(s). This paper is considered the first attempt to predict cycle-level through and left-turn movements at signalized intersections at corridor level using traffic data from upstream and downstream intersections in North/South direction. Each three consecutive intersections create one group. The objective is to predict through and left-turn movements at the middle intersection in each group. Thus, the 16 intersections were divided into six groups. The proposed approach sought to predict traffic movements at six intersections along two different corridors at once using one generic model.

The three models were trained, tuned and tested for through and left-turn movements' predictions. Hence, a total of 18 models were developed. The performance measures (MAE and RMSE) were calculated for each developed model and compared. The results showed that the performance measures for the three models were slightly close. However, the GRU model outperformed the other models in both through and left-turn movements' predictions. Also, the results concluded that the best performance was achieved when utilizing the dataset that contains six previous cycles; no further improvement occurred when increasing the number of utilized cycles to eight. The MAE and RMSE for the GRU model were 5.26 and 9.80 for through movements and 0.86 and 1.66 for left-turn movements, respectively.

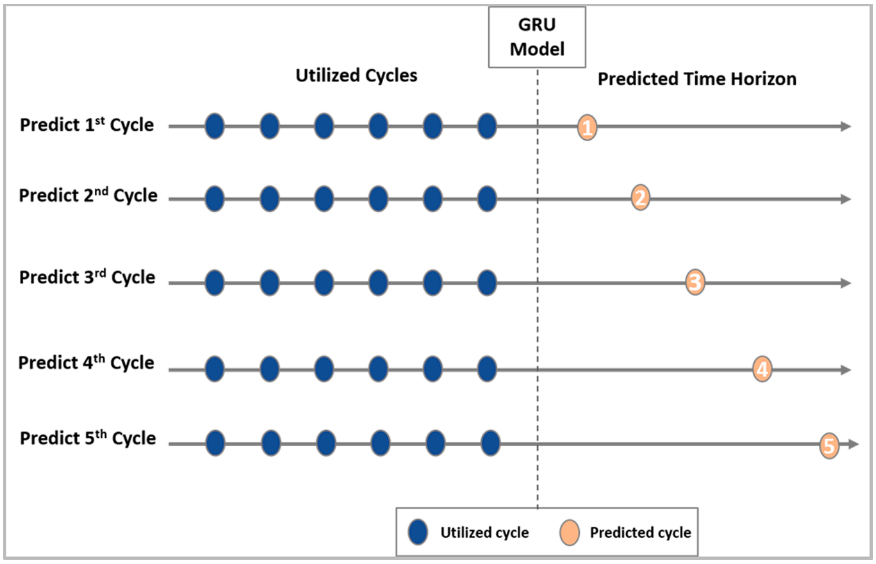

Fig. 6. Time horizon prediction architecture.

Table 4
Performance Measures for the Predicted Time Horizon.

Through Movement					
Developed Model	Predicted Cycle	Train Data		Test Data	
		MAE	RMSE	MAE	RMSE
GRU	1	4.26	8.11	5.26	9.80
	2	5.52	9.42	5.70	9.60
	3	5.64	8.88	5.75	9.62
	4	5.76	9.75	5.88	9.96
	5	5.47	9.60	5.89	9.99
Left-turn Movement					
Developed Model	Predicted Cycle	Train Data		Test Data	
		MAE	RMSE	MAE	RMSE
GRU	1	0.83	1.60	0.86	1.66
	2	0.85	1.64	0.88	1.72
	3	0.82	1.60	0.88	1.72
	4	0.85	1.65	0.89	1.73
	5	0.86	1.67	0.89	1.73

Afterwards, as the GRU modelling approach achieved the best performance, it was developed to predict through and left-turn movements for the next time horizons (up to 5 cycles ahead). Similarly, the dataset that contains six previous cycles were utilized in the time horizons prediction. Following the same approach, the GRU model was trained, tuned, and tested. Then, the same performance measures were computed. The results showed slight increase in MAE and RMSE when the number of predicted cycle increases. However, the performance measures for the 5 cycles were in the same range. Thus, the GRU model was able to predict through and left-turn movements for five cycles ahead with good performance measures. In conclusion, the adopted GRU model outperformed other proposed models in this paper as well as in the previous literature. Furthermore, the developed GRU model also provided accurate time horizon predictions for five cycles in the future. This could be useful in several applications that help in improving corridor management and safety.

CRediT authorship contribution statement

Nada Mahmoud: Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing - original draft, Visualization. **Mohamed Abdel-Aty:** Conceptualization, Methodology, Resources, Supervision, Project administration, Funding acquisition. **Qing Cai:** Conceptualization, Methodology, Resources. **Jinghui Yuan:** Methodology, Software, Resources.

Acknowledgments

The research discussed in this paper was funded by Florida Department of Transportation (FDOT).

The authors thank Amr Abdelraouf for his considerable help and valuable discussions in this study. This paper and its contents, including conclusions and results, are solely those of the authors; they do not represent opinions or policies of the funding agency.

References

Ahmed, M.S., Cook, A.R., 1979. Analysis of Freeway Traffic Time-Series Data By Using Box-Jenkins Techniques. Transp. Res. Rec. 1–9.

Bengio, Y., Boulanger-Lewandowski, N., Pascanu, R., 2013. ADVANCES IN OPTIMIZING RECURRENT NETWORKS. 2013 IEEE Int. Conf. Acoust. Speech Signal Process. Vancouver, BC 8624–8628. https://doi.org/10.1109/ICASSP.2013.6639349.

Cao, X., Zhong, Y., Zhou, Y., Wang, J., Zhu, C., Zhang, W., 2017. Interactive temporal recurrent convolution network for traffic prediction in data centers. IEEE Access 6, 5276–5289. https://doi.org/10.1109/ACCESS.2017.2787696.

Castillo, E., Menéndez, J.M., Sánchez-Cambronero, S., 2008. Predicting traffic flow using Bayesian networks. Transp. Res. Part B Methodol. 42, 482–509. https://doi.org/10.1016/j.trb.2007.10.003.

Chang, S.C., Kim, R.S., Kim, S.J., Ahn, M.H., 2000. Traffic-flow forecasting using a 3-stage model. Proc. IEEE Intell. Veh. Symp. 2000, 451–456. https://doi.org/10.1109/IVS.2000.898384.

Chen, A., Chootinan, P., Ryu, S., Lee, M., Recker, W., 2012. An intersection turning movement estimation procedure based on path flow estimator. J. Adv. Transp. 46, 161–176. https://doi.org/10.1002/atr.

Chen, M., Chien, S., 2001. Dynamic Freeway Travel-Time Prediction with Probe Vehicle Data: Link Based Versus Path Based. Transp. Res. Rec. J. Transp. Res. Board 1768, 157–161. https://doi.org/10.3141/1768-19.

Chen, T., Guestrin, C., 2016. XGBoost: A scalable tree boosting system. Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. 785–794 https://doi.org/10.1145/2939672.2939785.

Cheng, J., Li, G., Chen, X., 2019. Research on travel time prediction model of freeway based on gradient boosting decision tree. IEEE Access 7, 7466–7480. https://doi.org/10.1109/ACCESS.2018.2886549.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. EMNLP 2014 - 2014 Conf. Empir. Methods Nat. Lang. Process. Proc. Conf. 1724–1734. https://doi.org/10.3115/v1/d14-1179.

Clark, S., 2003. Traffic Prediction Using Multivariate Nonparametric Regression. J. Transp. Eng. 129, 161–168. https://doi.org/10.1061/(ASCE)0733-947X(2003)129:2(161).

Clark, S.D., Dougherty, M.S., Kirby, H.R., 1993. The use of neural networks and time series models for short term traffic forecasting: a comparative study. PTRC 21st Summer Annu. Meet. 151–162. https://doi.org/10.1007/978-1-4612-0533-3_7.

Cui, Z., Ke, R., Wang, Y., 2018. Deep Bidirectional and Unidirectional LSTM Recurrent Neural Network for Network-wide Traffic Speed Prediction. arXiv:1801.02143 1–11.

Davis, B.G.A., Nihan, N.L., 1991. Nonparametric regression and short-term freeway traffic forecasting. J. Transp. Eng. 117, 178–188.

Dia, H., 2001. An object-oriented neural network approach to short-term tra c forecasting. Eur. J. Oper. Res. 131, 253–261.

Dong, X., Lei, T., Jin, S., Hou, Z., 2018. Short-term Traffic Flow Prediction Based on XGBoost. IEEE 7th Data Driven Control and Learning Systems Conference. IEEE 854–859.

Friedman, J., Hastie, T., Tibshirani, R., 2000. Additive logistic regression: a statistical view of boosting (With discussion and a rejoinder by the authors). Ann. Stat. 28, 337–407. https://doi.org/10.1214/aos/1016218223.

Fu, R., Zhang, Z., Li, L., 2016. Using LSTM and GRU Neural Network Methods for Traffic Flow Prediction. IEEE 31st Youth Acad. Annu. Conf. Chinese Assoc. Autom. 324–328.

Ghanim, M.S., Shaaban, K., 2018. Estimating Turning Movements at Signalized Intersections Using Artificial Neural Networks. IEEE Trans. Intell. Transp. Syst. 1–9. https://doi.org/10.1109/TITS.2018.2842147.

Ghosh, B., Basu, B., O'Mahony, M., 2007. Bayesian time-series model for short-term traffic flow forecasting. J. Transp. Eng. 133, 180–189. https://doi.org/10.1038/010398d0.

Ghosh, B., Basu, B., O'Mahony, M., 2005. Time-series modelling for forecasting vehicular traffic flow in Dublin. 84th Transp. Res. Board Conf. Natl. Acad. Washingt. D.C. https://doi.org/10.1109/JPHOTOV.2015.2434597.

Graves, A., Mohamed, A.R., Hinton, G., 2013. Speech recognition with deep recurrent neural networks. ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc. 6645–6649 https://doi.org/10.1109/ICASSP.2013.6638947.

Greenwell, B., Boehmke, B., Cunningham, J., 2019. Generalized Boosted Regression Models.

GRIDSMART USERGUIDE, 2019.

Guo, F., Krishnan, R., Polak, J., 2013. A computationally efficient two-stage method for short-term traffic prediction on urban roads. Transp. Plan. Technol. 36, 62–75. https://doi.org/10.1057/s41309-017-0025-x.

Guo, F., Polak, J.W., Krishnan, R., 2018. Predictor fusion for short-term traffic forecasting. Transp. Res. Part C Emerg. Technol. 92, 90–100. https://doi.org/10.1016/j.trc.2018.04.025.

Guo, J., Huang, W., Williams, B.M., 2014. Adaptive Kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification. Transp. Res. Part C Emerg. Technol. 43, 50–64. https://doi.org/10.1016/j.trc.2014.02.006.

Hochreiter, S., Schmidhuber, J., 1997. Long Short-Term Memory. Neural Comput. 9, 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.

Jiang, X., Adeli, H., Asce, H.M., 2005. Dynamic Wavelet Neural Network Model for Traffic Flow Forecasting. J. Transp. Eng. 131, 771–779.

Kamarianakis, Y., Prastacos, P., 2005. Space-time modeling of traffic flow. Comput. Geosci. 31, 119–133. https://doi.org/10.1016/j.cageo.2004.05.012.

Koch, P., Wujek, B., Golovidov, O., Gardner, S., 2017. Automated Hyperparameter Tuning for Effective Machine Learning. Proc. SAS Glob Forum 2017 Conf. SASInstitute Inc 1–23.

Kong, Q.J., Xu, Y., Lin, S., Wen, D., Zhu, F., Liu, Y., 2013. UTN-model-based traffic flow prediction for parallel-transportation management systems. IEEE Trans. Intell. Transp. Syst. 1541–1547 https://doi.org/10.1109/TITS.2013.2252463.

Li, P., Abdel-aty, M., Yuan, J., 2020a. Real-time crash risk prediction on arterials based on LSTM-CNN ⋆. Accid. Anal. Prev. 135, 105371 https://doi.org/10.1016/j.aap.2019.105371.

Li, W., Ban, X., 2019. A Deep Learning Approach for Lane-Based Short-Term Traffic Volume Prediction at Signalized Intersections. Transportation Research Board 98th Annual Meeting.

Li, W., Wang, J., Fan, R., Zhang, Y., Guo, Q., Siddique, C., Jeff, X., 2020b. Short-term traffic state prediction from latent structures : Accuracy. Transp. Res. Part C 72–90. https://doi.org/10.1016/j.trc.2019.12.007.

Li, X., Bai, R., 2017. Freight Vehicle Travel Time Prediction Using Gradient Boosting Regression Tree. 2016 15th IEEE Int. Conf. Mach. Learn. Appl. 1010–1015. https://doi.org/10.1109/icmla.2016.0182.

Li, Z., Jiang, S., Li, L., Li, Y., 2019. Building sparse models for traffic flow prediction: an empirical comparison between statistical heuristics and geometric heuristics for Bayesian network approaches. Transp. B 7, 107–123. https://doi.org/10.1080/21680566.2017.1354737.

Lin, S., Xi, Y., Yang, Y., 2008. Short-term traffic flow forecasting using macroscopic urban traffic network model. IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC 134–138. https://doi.org/10.1109/ITSC.2008.4732567.

Liu, B., Cheng, J., Cai, K., Shi, P., Tang, X., 2017. Singular point probability improve LSTM network performance for long-term traffic flow prediction. National Conference of Theoretical Computer Science. 328–340.

Min, X., Hu, J., Chen, Q., Zhang, T., Zhang, Y., 2009. Short-term traffic flow forecasting of urban network based on dynamic STARIMA model. IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC 461–466. https://doi.org/10.1109/ITSC.2009.5309741.

Min, X., Hu, J., Zhang, Z., 2010. Urban traffic network modeling and short-term traffic flow forecasting based on GSTARIMA model. IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC 1535–1540. https://doi.org/10.1109/ITSC.2010.5625123.

Pan, B., 2018. Application of XGBoost algorithm in hourly PM2.5 concentration prediction. IOP Conference Series: Earth and Environmental Science. https://doi.org/10.1088/1755-1315/113/1/012127.

Qiao, W., Haghani, A., Hamedi, M., 2013. A nonparametric model for short-term travel time prediction using bluetooth data. J. Intell. Transp. Syst. Technol. Planning, Oper. 17, 165–175. https://doi.org/10.1080/15472450.2012.748555.

Rahman, R., Hasan, S., 2020. Real-time signal queue length prediction using long short-term memory neural network. Neural Comput. Appl. 9, 1–14. https://doi.org/10.1007/s00521-020-05196-9.

Rong, Y., Zhang, X., Feng, X., Ho, T.K., Wei, W., Xu, D., 2015. Comparative analysis for traffic flow forecasting models with real-life data in Beijing. Adv. Mech. Eng. 7, 1–9. https://doi.org/10.1177/1687814015620324.

Shoup, G., Remias, S.M., Hainen, A.M., Grimmer, G., Davis, A.D., 2013. Characterizing Reliability of Manual Intersection Turning Movement Counts Using Modern Data Collection Technology Characterizing Reliability of Manual Intersection Turning Movement Counts Using Modern Data Collection. JTRP Other Publ. Reports 6, 1–24.

Stathopoulos, A., Karlaftis, M.G., 2003. A multivariate state space approach for urban traffic flow modeling and prediction. Transp. Res. Part C Emerg. Technol. 11, 121–135. https://doi.org/10.1016/S0968-090X(03)00004-4.

Vlahogianni, E.I., Karlaftis, M.G., Golias, J.C., 2007. Spatio-temporal short-term urban traffic volume forecasting using genetically optimized modular networks. Comput. Civ. Infrastruct. Eng. 22, 317–325. https://doi.org/10.1111/j.1467-8667.2007.00488.x.

Vlahogianni, E.I., Karlaftis, M.G., Golias, J.C., 2005. Optimized and meta-optimized neural networks for short-term traffic flow prediction: A genetic approach. Transp. Res. Part C Emerg. Technol. 13, 211–234. https://doi.org/10.1016/j.trc.2005.04.007.

Williams, B.M., Hoel, L.A., 2003. Modeling and Forecasting Vehicular Traffic Flow as a Seasonal ARIMA Process: Theoretical Basis and Empirical Results. J. Transp. Eng. 129, 664–672. https://doi.org/10.1061/(ASCE)0733-947X(2003)129:6(664).

Wu, Y.J., Chen, F., Lu, C.T., Yang, S., 2016. Urban Traffic Flow Prediction Using a Spatio-Temporal Random Effects Model. J. Intell. Transp. Syst Technol. Planning, Oper. 20, 282–293. https://doi.org/10.1080/15472450.2015.1072050.

Xia, D., Wang, B., Li, H., Li, Y., Zhang, Z., 2016. A distributed spatial-temporal weighted model on MapReduce for short-term traffic flow forecasting. Neurocomputing 179, 246–263. https://doi.org/10.1016/j.neucom.2015.12.013.

Xia, Y., Chen, J., 2017. Traffic Flow Forecasting Method based on Gradient Boosting Decision Tree. Adv. Eng. Res. 130, 413–416. https://doi.org/10.2991/fmsmt-17.2017.87.

Xie, Y., Zhang, Y., Ye, Z., 2007. Short-term traffic volume forecasting using Kalman filter with discrete wavelet decomposition. Comput. Civ. Infrastruct. Eng. 22, 326–334. https://doi.org/10.1111/j.1467-8667.2007.00489.x.

Yang, B., Sun, S., Li, J., Lin, X., Tian, Y., 2019. Traffic flow prediction using LSTM with feature enhancement. Neurocomputing 332, 320–327. https://doi.org/10.1016/j.neucom.2018.12.016.

Yang, S., Wu, J., Du, Y., He, Y., Chen, X., 2017. Ensemble Learning for Short-Term Traffic Prediction Based on Gradient Boosting Machine. J. Sensors 2017, 1–15. https://doi.org/10.1155/2017/7074143.

Yoon, B., Chang, H., 2014. Potentialities of Data-Driven Nonparametric Regression in Urban Signalized Traffic Flow Forecasting. J. Transp. Eng. 140, 548–556. https://doi.org/10.1061/(ASCE)TE.1943-5436.

Zhan, H., Gomes, G., Li, X.S., Madduri, K., Sim, A., Wu, K., 2018. Consensus Ensemble System for Traffic Flow Prediction. IEEE Trans. Intell. Transp. Syst. 1–12.

Zhang, F., Zhu, X., Hu, T., Guo, W., Chen, C., Liu, L., 2016. Urban link travel time prediction based on a gradient boosting method considering spatiotemporal correlations. ISPRS Int. J. Geo-Information 5. https://doi.org/10.3390/ijgi5110201.

Zhang, W., Yu, Y., Qi, Y., Shu, F., Wang, Y., 2019. Short-term traffic flow prediction based on spatio-temporal analysis and CNN deep learning. Transp. A Transp. Sci. 15, 1688–1711. https://doi.org/10.1080/23249935.2019.1637966.

Zhang, Y., Haghani, A., 2015. A gradient boosting method to improve travel time prediction. Transp. Res. Part C 58, 308–324. https://doi.org/10.1016/j.trc.2015.02.019.

Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., Deng, M., Li, H., 2019. T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction. IEEE Trans. Intell. Transp. Syst. 1–11 https://doi.org/10.1109/tits.2019.2935152.

Zhao, Z., Chen, W., Wu, X., Chen, P.C.V., Liu, J., 2017. LSTM network: A deep learning approach for short-term traffic forecast. IET Image Process. 11, 68–75. https://doi.org/10.1049/iet-its.2016.0208.

Zheng, J., Liu, H.X., 2017. Estimating traffic volumes for signalized intersections using connected vehicle data. Transp. Res. Part C 347–362.

Zheng, W., Lee, D.-H., Shi, Q., 2006. Short-Term Freeway Traffic Flow Prediction: Bayesian Combined Neural Network Approach. J. Transp. Eng. 132, 114–121. https://doi.org/10.1061/(ASCE)0733-947X(2006)132:2(114).

Zheng, Z., Su, D., 2014. Short-term traffic volume forecasting: A k-nearest neighbor approach enhanced by constrained linearly sewing principle component algorithm. Transp. Res. Part C Emerg. Technol. 43, 143–157. https://doi.org/10.1016/j.trc.2014.02.009.

Zheng, Z., Yang, Y., Liu, J., Dai, H.N., Zhang, Y., 2019. Deep and Embedded Learning Approach for Traffic Flow Prediction in Urban Informatics. IEEE Trans. Intell. Transp. Syst. 20, 3927–3939. https://doi.org/10.1109/TITS.2019.2909904.

Zhu, J.Z., Cao, J.X., Zhu, Y., 2014. Traffic volume forecasting based on radial basis function neural network with the consideration of traffic flows at the adjacent intersections. Transp. Res. Part C Emerg. Technol. 47, 139–154. https://doi.org/10.1016/j.trc.2014.06.011.

Zhu, Z., Peng, B., Xiong, C., Zhang, L., 2016. Short-term traffic flow prediction with linear conditional Gaussian Bayesian network. J. Adv. Transp. 50, 1111–1123. https://doi.org/10.1002/atr.