Real-time Traffic Data Prediction with Basic Safety Messages using Kalman-Filter based Noise Reduction Model and Long Short-term Memory Neural Network

Mizanur Rahman*, Ph.D.
Postdoctoral Fellow
Center for Connected Multimodal Mobility (C²M²)
Glenn Department of Civil Engineering, Clemson University
127 Lowry Hall, Clemson, SC 29634
Tel: (864) 650-2926; Email: mdr@clemson.edu

Mashrur Chowdhury, Ph.D., P.E., F.ASCE
Eugene Douglas Mays Endowed Professor of Transportation
Glenn Department of Civil Engineering, Clemson University
216 Lowry Hall, Clemson, SC 29631
Tel: (864) 656-3313; Email: mac@clemson.edu

Jerome McClendon, Ph.D.
Research Assistant Professor
Automotive Engineering
4 Research Drive, Greenville, SC 29607
Tel: (334) 538-7539; Email: JMCCLEN@g.clemson.edu

*Corresponding Author
ABSTRACT

The accurate prediction of traffic data, such as average speed and average space-headway between vehicles in real-time is important for route planning and scheduling to reduce travel time, future traffic condition assessment, and for vehicle’s energy optimization to reduce fuel consumption. Unfortunately, the stochastic change of traffic flow over time greatly complicates the development of such a real-time traffic data prediction method. With the development of Connected Vehicle (CV) technology, temporal variation of roadway traffic can be captured by sharing Basic Safety Messages (BSMs) from each vehicle using the communication between vehicles as well as with transportation roadside infrastructures (e.g., traffic signal, roadside unit) and traffic management centers. However, the penetration of connected vehicles in the near future will be limited. BSMs from limited CVs could provide an inaccurate estimation of current speed or space-headway. This inaccuracy in the estimated current average speed and average space-headway data is termed as noise. This noise in the traffic data significantly reduces the prediction accuracy of a machine-learning model, such as the accuracy of long short-term memory (LSTM) model in predicting traffic condition. To improve the real-time prediction accuracy with low penetration of CVs, we developed a traffic data prediction model that combines the LSTM with a noise reduction model. We first investigated the standard Kalman filter and Kalman filter based Rauch–Tung–Striebel (RTS) noise reduction techniques to reduce the noise from the current traffic data measured from BSMs. We next used the filtered data to evaluate the performance of the LSTM prediction model. The average speed and space-headway used in this study were generated from the Enhanced Next Generation Simulation (NGSIM) dataset, which contains vehicle trajectory data for every one-tenth of a second. Compared to a baseline LSTM model without any noise reduction, for 5% penetration of CVs, the analyses revealed that combined LSTM/RTS model reduced the mean absolute percentage error (MAPE) from 19% to 5% for speed prediction and from 27% to 9% for space-headway prediction. The overall reduction of MAPE value ranged from 1% to 14% for speed and 2% to 18% for space headway prediction compared to the baseline model. The statistical significance test with a 95% confidence interval confirmed no significant difference in predicted average speed and average space headway using this LSTM/RTS combination for different CV penetration rates.

Keywords: Connected vehicles, traffic data prediction, machine learning, Kalman filter, long short-term memory.
1. INTRODUCTION

The evolution of new Intelligent Transportation Systems (ITS) technologies has made possible transportation initiatives such as Advanced Traffic Management Systems (ATMS) and Advanced Traveler Information Systems (ATIS) to inform travelers about current and future traffic conditions (Vanajakshi and Rilett, 2004; Ma et al., 2012; Khan et al., 2017). These traffic management strategies, which depend upon the accurate prediction of travel speed and space headway between vehicles are used for route planning and scheduling to reduce travel time, for future traffic condition assessment, and for energy optimization to reduce fuel consumption (Ma et al., 2015; Hua et al., 2017; Ren et al. 2017; Ma et al., 2009; Ma et al., 2012; Khan et al., 2017).

The stochastic change of traffic flow over time, depending on time of day or day of week greatly complicates the development of such a real-time traffic data prediction methodology (Ma et al., 2015). As such, capturing the temporal relationship over time to predict traffic data accurately is most important. Currently, these data are collected through a wide range of such roadway traffic sensors as inductive loop detectors and video cameras. The problem is that such data cannot simultaneously capture the stochastic nature of the traffic flow in terms of spatial and temporal variation for a specific segment of a roadway as these sensors are deployed at a fixed location (Ma et al., 2015; Zhao et al., 2017).

These issues are currently being addressed through the use of Connected Vehicle (CV) technologies that provide interconnection between transportation systems allowing vehicles to share Basic Safety Messages (BSMs) by communicating with one another as well as with transportation roadside infrastructures (e.g., traffic signal, roadside unit) and Traffic Management Centers (TMC). In this CV system, BSMs provide trajectory data, such as the location, speed, acceleration and deceleration, of each vehicle for every one-tenth of a second (Liu and Khattak, 2016; Du et al., 2017). Providing real-time BSMs with this temporal variation requires data-driven approaches, such as machine learning models for capturing the non-linearity of traffic patterns.

More specifically, recurrent neural networks (RNNs), a type of machine learning model that can capture temporal variation and predict time series data, have been used to predict freeway traffic volume (Zhao et al., 2017). However, these traditional RNNs are unable to capture the long temporal dependency in the traffic patterns due to an unexpected traffic event on a roadway, such as a traffic incident that occurs one hour previously may still cause severe congestion in the subsequent two to three hours after the event (Zhao et al., 2017; Ma et al., 2015). To address these limitations, a special RNN architecture, the Long Short-Term Memory (LSTM) neural network (Hochreiter and Schmidhuber, 1997) has been developed for time series prediction.

Over the past decade, the LSTM has been successfully used in (i) robot control; (ii) speech recognition; (iii) handwriting recognition; and (iv) human action recognition (Zhao et al., 2017). It has also been used to predict univariate traffic data, such as volume and speed, using data collected from roadway sensors (Zhao et al., 2017; Ma et al., 2015). However, the near absence of research using LSTM and BSMs from connected vehicles to predict traffic data limits the near future penetration of connected vehicles, which prevents accurate estimation of current speed or space headway. We define this inaccuracy in the estimated current speed and space-headway data as noise. The noise in the traffic data can significantly reduce the prediction accuracy of a machine-learning model, such as long short-term memory (LSTM). In addition, a massive amount of data including a deep learning model is required to accurately predict speed and space headway with noisy traffic data and to achieve an expected prediction accuracy.

To improve the real-time prediction accuracy with low penetration of CVs, we developed a traffic data prediction model that combines the LSTM with a noise reduction model. We first
investigated two noise reduction models, the standard Kalman filter and the Kalman filter based Rauch–Tung–Striebel (RTS) data smoothing techniques, to reduce the noise from the traffic data measured from BSMs. Second, the performance of the LSTM prediction model was evaluated for predicting traffic data using the resulting filtered data. Using a vehicle penetration rate ranging from 5% to 90%, Enhanced Next Generation Simulation (NGSIM) data (Montanino and Punzo, 2013), which contain vehicle trajectory data for every one-tenth of a second, was used as the BSMs for the evaluation of the LSTM prediction model.

The remainder of this paper is structured as follows. Section 2 describes the related work on traffic data prediction using machine learning and noise reduction models, while Section 3 discusses the analysis of traffic data from BSMs of CVs in a mixed traffic scenario (connected and non-connected vehicles) to explore the noise resulting from the speed and space headway data. Section 4 analyzes a method for predicting traffic data using different noise reduction models at low penetration of CVs, with the evaluation of the resulting method being discussed in Section 5 along with the analysis results. Section 6 discusses the real-time application efficacy and Section 7 provides a concluding discussion.

2. RELATED WORK

The related work analyzes existing research on different types of recurrent neural network models for traffic prediction and noise reduction models.

2.1 Recurrent Neural Network Models for Traffic Data Prediction

Feed Forward Neural Networks (FFNN), the simplest neural networks, have been used in forecasting travel time and traffic flow as well as subsequent traffic patterns (Park and Rilett, 1999; Chowdhury et al., 2006; Ma et al., 2009). However, they are unable to capture temporal and spatial variations in time series problems, as they do not have a memory mechanism, which can recall the effect of dynamic nature of traffic patterns, for purposes of mapping future traffic flow predictions. In addition, spatial and temporal patterns and optimal look-back intervals must be determined prior to input into FFNN for the time series prediction, a step requiring data preprocessing using statistical methods (e.g., correlation analyses, principal component analysis, and genetic algorithm) to prepare a large enough time series dataset to capture spatial and temporal patterns that is not efficient for real-time time series predictions.

More recently, RNNs have been explored for capturing variations over time for a time series problem including, for example the Time-Delay Neural Network (TDNN), the Jordan–Elman Neural Network, and the State-Space Neural Network (SSNN)). The first two models have been applied to traffic speed predictions using 30-s loop-detector speed data from a freeway segment of Interstate 4 in Orlando, Florida, the results showing that these models outperformed non-linear statistical time series model (Ishak et al., 2003). The SSNN model has also been used for real-time short-term freeway travel time prediction using synthetic and real-world data (Van Lint et al., 2002, 2005; Liu et al., 2006), one example being the real-time data collected from freeway and urban scenarios for the Regiolab-Delft Project (Van Lint et al. 2005; Van Zuylen and Muller 2002). However, these RNN models are unable to capture temporal and spatial relationship for a long-term time series problem because of vanishing gradient and exploding gradient problems.

To address these issues for long-term time series problems, the LSTM model has been explored for traffic data prediction. For example, Ma et al. (2015) used a three-hidden-layer LSTM model for traffic speed prediction utilizing microwave sensor data. The hidden layer of this model
includes a memory block for capturing the non-linear patterns of speed over the time. This research found that the LSTM provided more accurate predictions than traditional RNN models by determining optimal time lags using a trial and error method. More recently, Zhao et al. (2017) constructed a multi-layers LSTM network for traffic volume prediction. Their model includes an Origin-Destination Correlation (ODC) matrix integrated in the LSTM network. This ODC matrix captures the correlations between the temporal and spatial patterns among different links of a road network, thus improving the performance of the LSTM model by capturing traffic flow evolution over time and space. This study found that the two-dimensional LSTM was more accurate than existing traffic forecast methods for short-term travel speed prediction. In further study, Wang et al. (2017) developed a deep neural network using an LSTM for predicting driver behavior. As driver behavior is a time-dependent phenomenon and an LSTM can mimic human memory, they developed an LSTM-based car-following model that can replicate driver behavior using microscopic NGSIM data. This research found that this deep neural network model exhibits significantly higher accuracy than existing car-following models.

However, the use of an LSTM has yet to be undertaken to predict multivariate traffic data using BSMs in a connected vehicle environment at a low penetration rate of CVs. As the related work indicates that this type of RNN has the capability of capturing long-term dependency for predicting time series data, this study used an LSTM model for predicting traffic data in a connected vehicle environment. In this environment, the LSTM model can learn non-linear time-variant traffic behavior from a training data set and predict traffic data based on the real-time input of traffic data. However, this learning capability of the LSTM model can be reduced by the noise in the data from a mixed traffic environment (i.e., connected and non-connected vehicles), as one cannot expect 100% CVs in the near future.

2.2 Noise Reduction Models

Noise reduction models have been used extensively to analyze given measurements and to estimate accurate measurements because of inaccuracies of sensor collected data. Previously vehicle trajectories data were filtered using the following methods: (i) averaging (Ossen and Hoogendoorn 2008); (ii) locally weighted regression using the tri-cube weight function (Toledo et al. 2007); (iii) filtering (Punzo et al. 2005; Montanino and Punzo 2013) and (iv) moving average techniques (Thiemann et al. 2008). The noise reduction accuracy of these methods depends on a window size. Kanagaraj et al. (2015) and Rim et al. (2016) used locally weighted regression techniques for smoothing erroneous vehicle coordinates and speed data, respectively, both finding that the accuracy of locally weighted regression varies based on the polynomial order. More recently, Punzo et al. (2005, 2011) used moving average and low pass filtering techniques to correct GPS-based trajectory data, with the latter study analyzing the vehicle trajectory and speed data and evaluating the accuracy in terms of jerk, consistency, and spectral analysis. They found that the low pass filter performs very well in terms of accuracy.

Another widely used data smoothing technique is the Kalman Filter, which is used to reduce noise from sensor fault in sensor-collected data. This filter, named after Rudolf E. Kalman, who provided the concept for this method (Kalman, 1960), estimates the current state based on a sequence of previous noisy observations. There are three types of Kalman filter smoothing, fixed-interval smoothing, fixed-point smoothing, and fixed-lag smoothing in addition to several variations including the standard Kalman filter, the extended Kalman filter and the scented Kalman filter (Gadsden and Lee, 2017). If the noise in the sensor-collected data is Gaussian, the standard Kalman filter is applicable for the noise reduction. The standard Kalman Filter has been found to
be effective in estimating air vehicle sensor errors. In the past, Ervin et al. (1991) used a Kalman filter to smooth the vehicle trajectory data.

Using the standard Kalman filter process, Rauch et al. (1965) developed an efficient method based on the RTS algorithm, a two-pass algorithm that reduces the computational effort required for Kalman filter smoothing since it requires the standard Kalman filter to be implemented only in the forward direction. The forward pass is the standard Kalman filter while the backward recursion is introduced to reduce the inherent bias in the Kalman filter estimates. Based on their applicability, the RTS and standard Kalman filter have been implemented as noise reduction models. In a mixed traffic scenario (connected and non-connected vehicles), data collected from the low penetration of connected vehicles (e.g., 5% CVs, 10% CVs) in addition to the temporal variation of traffic make the data noisy.

3. ANALYSIS OF TRAFFIC DATA WITH DIFFERENT PENETRATION OF CVs

This section describes the data collected from each vehicle of a 500m (1650 ft) roadway section on Interstate 80 segment in Emeryville (San Francisco), California (NGSIM, 2006), followed by analysis for 10 different penetration rates to determine the noise at the low penetration of CVs.

3.1 Basic Safety Messages (BSMs) Obtained from the Enhanced NGSIM Dataset

The original NGSIM dataset, which was collected through video cameras, represents 45 minutes of the peak afternoon period, specifically 4:00 PM to 4:15 PM, and 5:00 PM to 5:30 PM (NGSIM, 2006). The vehicle trajectory data of each vehicle were generated through video image processing. However, since the original NGSIM data contain inconsistencies and noise, Montanino and Punzo et al. (2013) improved the data set using a multistep procedure to reconstruct the original I 80 dataset (4:00 PM to 4:15 PM) for each vehicle trajectory, and subsequently conducted an extensive exploratory study to determine the accuracy of NGSIM trajectory data (Punzo et al., 2011). They reconstructed the original data measurements while preserving: i) the dynamics of the actual vehicle while being driven (i.e., shifting gears, vehicle stoppages); ii) the internal vehicle trajectory consistency (i.e., the vehicle trajectory consistency for acceleration/deceleration, speed, and space travelled); and iii) the consistency in the platoon (i.e. the actual space headway between the follower and leader vehicles in the traffic stream) (Montanino and Punzo, 2013). This reconstructed data set is referred to as the Enhanced NGSIM dataset.

As the Enhanced NGSIM dataset was collected with a frequency of one-tenth of a second, it represents a sample of the BSMs (i.e., Vehicle ID, Timestamp, Lane ID, Location, Acceleration/Deceleration, Vehicle Length, Vehicle Class ID, Follower Vehicle ID, and Immediate Preceding Vehicle ID) generated from a connected vehicle environment. The study reported here used trajectory data from 3,335 vehicles with a frequency of 10 Hz from the Enhanced NGSIM I80-1 dataset. More specifically, the following subsection section explored the noise and outliers in the speed and space headway data from 5% to 90% penetration of CVs (5%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, and 90%).

3.2 Identification of Noise in the Speed and Space Headway Data

The BSMs in this study include time stamp, location coordinate, speed, acceleration/deceleration, relative speed, lane number, leader vehicle number and follower vehicle number. To prepare the data as a time series problem, this study used the Frame ID sequence from the NGSIM data as each video frame was created every one-tenth of a second. Then, using the location coordinate of each vehicle, the space headway of a vehicle was calculated. Average speed and space headway
time series data extracted from BSMs at different penetration rates of CVs were used as the input for the noise reduction model. Low penetration of CVs would very likely be unable to provide accurate average speed and average space headway data of all vehicles on a roadway segment. Data from limited CVs provide an inaccurate estimation of current average speed or space headway, which contributes to having a large number of outliers in the average traffic data. Figures 1(a) and 1(b) show the box plots of the average speed and space headway data, respectively, with varying penetrations of CVs. The number of outliers increases in average speed and average space-headway data with decreasing penetration rates of CVs, also indicating that the speed and headway data change drastically over time (as shown in Figure 1).

To observe the drastic change in traffic data over time, we compared the average speed and space headway profile between 100% penetration of CVs and 10% penetration of CVs in Figures 2(a) and 2(b), respectively, as an example. As shown in Figures 2(a) and 2(b), the average speed and space-headway change drastically over time with 10% penetration of CVs compared to 100% penetration of CVs. As this comparison indicates, both the speed and space headway estimation from the low penetration of CVs is inaccurate. We define this inaccuracy in the estimated current speed and space headway from a limited number of CVs as a noise. This noise leads to inaccurate prediction of traffic data using machine-learning models. In addition, a massive amount of data including a deep learning model is required to develop a prediction model with this noisy traffic data to capture the variation of the traffic behavior and to achieve expected accuracy. However, it is possible to achieve an expected prediction accuracy with low penetration rate of CVs if we can reduce the noise in the traffic data.

We identify the type of the noise distribution to reduce noise (i.e., inaccuracy) in the estimated current traffic data, as the selection of noise reduction model depends on the distribution of noise. For example, the Standard Kalman Filter and RTS noise reduction model can be used if the noise distribution in the data is Gaussian. Figures 2(c) and 2(d) present the histogram of the noise distributions in the speed and space headway data. To analyze the type of noise distributions, the noise for each observation was calculated by subtracting the speed or space headway data for the 10% penetration from the 100% penetration of CVs. This process was followed for all ten penetration rates studied here. The histogram plots of the noise distributions indicate that all distributions look normally distributed (a Gaussian distribution). In addition, the analysis of the distributions indicates that the noise distributions for all penetration rates followed the similar distribution.
Fig. 2. Identifying noise in speed and space headway distribution for 10% penetration of CVs: (a) Comparison between 10% penetration of CVs and 100% CVs for speed data; (b) Comparison between 10% penetrations of CVs and 100% CVs for space headway data; (c) Histogram of noise distributions in speed data for 10% CVs penetration; and (d) Histogram of noise distributions in space headway data for 10% CVs penetration.
To confirm the normality (Gaussian distribution) of the noise, the data were further analyzed using a quantile-quantile plot (Q-Q plot). In this study, Q-Q plot generated idealized samples based on the Gaussian or normal distribution from the given noise. These idealized samples were divided into groups called quantiles. Each data point in the sample was paired with a similar member from the idealized distribution at the same cumulative distribution, and the resulting points were plotted as a scatter plot with the idealized value on the x-axis and the data sample on the y-axis. The resulting plot indicated that the idealized samples followed the normal distribution lines, confirming that the noise distributions for all the penetration rates followed a normal distribution. Figure 3 presents an example of one of the Q-Q plots generated for noise in the average speed and space headway data.

![Idealized samples](image1)

(a) Noise in the speed data for 10% penetration of CVs

![Idealized samples](image2)

(b) Noise in the space headway data for 10% penetration of CVs

Fig. 3. An example of one of the Q-Q plots generated for noise in the average speed and space headway data

4. TRAFFIC DATA PREDICTION MODEL DEVELOPMENT

The general framework for the traffic data prediction model using LSTM combined with the noise reduction model is presented in Figure 4. As enhanced NGSIM data were collected every one-tenth of a second for each vehicle, the data were used as a part of the BSMs in a connected vehicle environment. Noise reduction models were used to filter noise from average traffic data. After the filtered data from the noise reduction model were put into a temporal sequence, training and testing data were prepared. Using the normalized training data, the optimal hyperparameter set of the LSTM model was determined. The testing data were then used to evaluate the traffic data prediction model.
Basic Safety Messages (BSMs) considered in this study
BSMs: time stamp, latitude, longitude, speed, acceleration/deceleration, relative speed, lane number, leader vehicle number, follower vehicle number

Data preparation for different penetrations of CVs

Investigation of noise in traffic data at different penetrations of CVs

Identification of noise reduction models for traffic data

LSTM model development for traffic data prediction

Performance evaluation of the prediction model

Fig. 4. Traffic data prediction method using LSTM combined with a noise reduction model.

4.1 Noise Reduction Model

We used the Standard Kalman filter and the Kalman filter based RTS noise reduction models to filter the noise in the average speed data. We also used the identical procedure using these two models for the space headway filtering.

4.1.1 Standard Kalman Filter

Based on the Kalman filter (Kalman, R. E., 1960), the state of speed $x_t$ at time $t$ evolves from the state $(x_{t-1})$ at $(t-1)$.

$$x_t = A_t x_{t-1} + B_t u_t + w_t$$

where

$A_t$ is the state transition matrix that transforms state $x_{t-1}$ to state $x_t$;

$B_t$ is the control-input matrix for measuring the correction for external influences based on control vector $u_t$;

$w_t$ is the process noise, which is assumed to be drawn from a zero mean multivariate normal distribution $N$ with covariance $Q_t$: $w_t \sim N(0, Q_t)$

At time $t$, $z_t$ is a measurement value calculated based on the linear combination of the new estimated speed $x_t$ and the measurement noise $v_t$. 
\[ z_t = H_t x_t + v_t \]

where

\( H_t \) is the measurement matrix that transforms the new estimated state of speed to a measured state

\( v_t \) is the measurement noise, which is assumed to be zero-mean Gaussian white noise with covariance \( R_t \): \( v_t \sim N(0, R_t) \)

The Kalman filter algorithm consists of two stages for reducing the noise of the speed data: i) prior estimation of the new state and ii) measurement update. Using the following equations, new speed are estimated at time \( t \). Here, \( P_t \) is a prior or posterior error covariance matrix, which measures the estimated accuracy of the state estimate.

\[
\hat{x}_{t|t-1}^{\text{prior}} = A_{t-1} \hat{x}_{t-1|t-1} + B_t u_t, \\
P_{t|t-1}^{\text{prior}} = A_{t-1} P_{t-1|t-1}^{\text{prior}} A_{t-1}^T + Q_t
\]

Next, the prior estimation of speed \( \hat{x}_{t-1|t}^{\text{prior}} \) and covariance \( P_{t-1|t}^{\text{prior}} \) are required for updating the measurement at time \( t \). Then the current speed \( \hat{x}_t \) can be estimated at time \( t \): 

\[
\hat{x}_t = \hat{x}_{t|t-1}^{\text{prior}} + K_t \left( z_t - H_t \hat{x}_{t|t-1}^{\text{prior}} \right)
\]

where \( K_t \) is the Kalman gain. Covariance \( P_t \) is calculated for updating the value of \( x \) at time \( t-1 \) as follows:

\[
P_t = \left( I - H_t K_t \right) P_{t|t-1}^{\text{prior}}
\]

Using the prior covariance \( P_t^{\text{prior}} \), the Kalman gain can be calculated as follows:

\[
K_t = P_{t|t-1}^{\text{prior}} H_t^T \left( H_t P_{t|t-1}^{\text{prior}} H_t^T + R_t \right)^{-1}
\]

### 4.1.2 Kalman Filter Based Rauch–Tung–Striebel (RTS) Model

The Rauch–Tung–Striebel (RTS) smoother uses the same forward pass as the standard Kalman filter algorithm (Rauch et al., 1965). The resulting prior and posterior speed estimates \( \hat{x}_{t|t-1} \) and \( \hat{x}_{t|n} \), and covariances \( P_{t|t-1} \) and \( P_{t|n} \) from the forward pass are used in the backward pass, which computes the smoothed speed estimates \( \hat{x}_{t|n} \) and covariance \( P_{t|n} \) (where \( t<n \)). To do so, the backward steps can be completed using the following recursive equations:
\[
\hat{x}_{t|n} = \hat{x}_{t|n} + C_t \left( \hat{x}_{t+1|n} - \hat{x}_{t+1|t} \right)
\]
\[
P_{t|n} = P_{t|n} + C_t \left( P_{t+1|n} - P_{t+1|t} \right) C_t^T
\]
Where,
\[
C_t = P_{t|n} A_{t+1} P_{t+1|n}^{-1}
\]
where \(x_{t|n}\) is the posterior speed estimate of time step \(t\); \(x_{t+1|n}\) is the prior speed estimate of time step \(t+1\); \(P_{t|n}\) is the posterior covariance estimate of time step \(t\), and \(P_{t+1|n}\) is the prior covariance estimate of time step \(t+1\).

### 4.2 Preparation of the Training and Testing Dataset

After processing the BSMs from the enhanced NGSIM dataset, the time series of speed and space headway data were used as input for a supervised learning problem. Specifically, the observation of speed or space headway at the current time step was used as an input to predict the traffic observation at the next time step. It should be noted that for prediction, the data were normalized between 0 and 1 before its use as an input of the LSTM model. After predicting the traffic data, the Root Mean Square Error (RMSE) was calculated based on the difference between the scaled value of predicted traffic data and the scaled value of actual traffic data (ground truth). To develop and evaluate the LSTM model, the dataset was divided into 1) training and 2) testing datasets. The training dataset contained 7000 samples, and the testing dataset 2800 samples for a total of 9800 samples.

### 4.3 Long Short-Term Memory (LSTM)

The LSTM model used in this research consists of (i) an input layer, (ii) a recurrent hidden layer, and (iii) an output layer (Hochreiter and Schmidhuber, 1997). The input sequence for the input layer is denoted as \(x = (x_1, x_2, x_3, \ldots, x_t)\), and the output sequence for the output layer is denoted as \(h = (h_1, h_2, h_3, \ldots, h_t)\), where \(t\) is the prediction period. In the context of speed and space headway prediction, \(x\) can be considered as current speed or space headway data, and \(h\) is the predicted speed. Of these layers, the primary layer is the recurrent hidden layer, which consists of a memory block, which solves the vanishing gradient (i.e., a change in the current speed or space headway causes very small change of the predicted speed or space headway) or exploding gradient (i.e., a change in the current speed or space headway causes very big change of the predicted speed or space headway) problems of traditional RNNs.
The memory block consists of (a) a forget gate, (b) an input gate, and (c) an output gate (as shown in Figure 5), all three of which control what information needs to be removed or added from the previous cell state to the new cell state. The input gate controls the activations of input into the memory block. The input gate $i_t$ decides which values need to be updated using a sigmoid activation function:

$$i_t = \text{sigmoid}(w_i x_i + u_i h_{t-1} + b_i)$$

where

\[ w, u = \text{parameter matrices} \]
\[ b = \text{bias} \]

The forget gate determines the information that must be forgotten from the previous cell state. Using a sigmoid layer, the forget gate layer $f_t$, which is represented by the following equation, determines the information to forget.

$$f_t = \text{sigmoid}(w_f x_i + u_f h_{t-1} + b_f)$$

Based on the input gate and forget gate information, the previous cell state, $c_{t-1}$, is updated to the new cell state $c_t$. To obtain the new cell state, the previous cell state $c_{t-1}$ is multiplied by $f_t$ to forget unnecessary information from the previous state. Then, new candidate values $i_t \odot \text{tanh}(w_i x_i + u_i h_{t-1} + b_i)$ are added to define how much is needed to update each state value:
The output gate controls the activations of output into the memory block. At the output gate, a sigmoid layer decides what parts of the cell state to output, $o_t$:

$$
o_t = \text{sigmoid} \left( w_o x_t + u_o h_{t-1} + b_o \right)
$$

Then, cell state $c_t$ is put through $\tanh$ (to push the values to between $-1$ and $1$) activation functions and multiplied by the output of the sigmoid gate output $o_t$ to predict speed or space headway $h_t$:

$$
h_t = o_t \odot \tanh(c_t)
$$

However, the prediction accuracy of the LSTM model depends on the determination of the optimal hyperparameter that includes the number of neurons, the number of epochs, the batch size, the dropout rate, and the learning rate.

### 4.4 Optimal LSTM Hyperparameter Determination

For the time series problem, traditional hyperparameter selection methods, such as the grid search method and the random search method (Bergstra and Bengio, 2012), are inapplicable for determining the optimal hyperparameter set. Thus, a trial and error procedure and the Root Mean Square Error (RMSE) metric was used to determine the optimal LSTM hyperparameter set. RMSE measures the square root of the average of the squared errors, which quantifies the difference between the predicted values and the actual values. The mathematical formulation of RMSE is as follows:

$$
\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2}
$$

where

- $N$ represents the total sample size
- $y_i$ is the actual value of traffic data (speed or space headway) and
- $\hat{y}_i$ is the predicted value of traffic data (predicted speed or predicted space headway).

A box and whisker plot was used to identify the optimal parameter set for a specific hyperparameter and compare the distribution of the RMSE scores for the various hyperparameter values. Figure 6 shows the box and whisker plot for the number of neurons selection process for the LSTM model for 5% penetration of CVs. Thirty samples for each value of a hyperparameter were created and subsequently plotted in the box and whisker plot to select the optimal hyperparameter set. The plot shows the median (green line), 25th and 75th percentiles of the data. This comparison also indicates that the optimal number of neurons is 100. However, the plot also shows that we could achieve better performance at the cost of worse performance on average. A similar procedure was followed for the selection of other hyperparameters (i.e., number of epoch, batch size, dropout rate, learning rate) for different penetrations of CVs. The optimal hyperparameter values for LSTM model are number of epochs = 400; number of neurons = 100; batch size = 50; dropout rate = 0.2; and learning rate = 0.001.
To train the LSTM model, we used a stochastic gradient descent algorithm with adaptive learning rate tricks (Kingma and Ba, 2014) and an optimal hyperparameter estimated as described in the previous section. The advanced gradient descent algorithm, ADAM, is an extension of the existing stochastic gradient descent algorithm (Kingma and Ba, 2014), which was used because of its applicability in natural language processing applications. After training the model using optimal hyperparameters and ADAM, we then determined the goodness of fit of the LSTM model to that of the testing datasets by determining the overfitting and underfitting problems to ensure the predictive capability of the model. A goodness of fit of a model is represented by an identical error that occurs in both the training and testing of the prediction model. An underfit model is represented by less error on the testing dataset than on the training dataset, and an overfit model is represented by less error on the training data which continuously improves, and which in terms of the testing set plateaus and then begins to degrade. If the loss (or error) on training and testing datasets decreases and stabilizes around the same point, then the model exhibits a good fit with the training and testing data. To assess the goodness-of-fit of the model using the training and testing datasets, the Mean Absolute Error (MAE), defined as the average of the absolute error, is used as a measurement of the loss (i.e., error) and as a metric to evaluate the performance. The mathematical formulation of MAE is given below:

\[
MAE = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i|
\]

where

- \(N\) represents the total sample size
- \(y_i\) is the actual value of traffic data (speed or space headway) and
- \(\hat{y}_i\) is the predicted value of traffic data (predicted speed or predicted space)

We plotted MAE profiles for training and testing dataset of different penetration of CVs to evaluate the goodness-of-fit of the LSTM model. Figure 7 shows the MAE profiles using optimal hyperparameters, and both the training and testing datasets from 5% to 30% penetration of CVs. Based on the comparison of MAE values of these two datasets, each model exhibits a good fit with the optimal hyperparameters.
Fig. 7. Comparison of Mean Absolute Error (MAE) profiles using training and testing datasets with the optimal parameter set.
5. EVALUATION RESULTS AND DISCUSSIONS

As the evaluation of the traffic data prediction model was based on the performance of the LSTM, this study used the filtered data resulting from the noise reduction models to predict the speed and space headway. Figure 8 compared between average speed profile (and space headway profile) with 100% CV penetration rates and filtered speed profile (and space headway profile) with 10% CV penetration rates using standard Kalman Filter and RTS filters. We observed that RTS filter was performing better than standard Kalman filter to denoise data of 10% CV penetration rates. We also found that the filtered average speed and space headway profiles with 10% penetration of CVs are following the trend of the profiles with 100% penetration of CVs very closely. The performance of these filters was quantitatively evaluated through the prediction accuracy of average speed and space headway using the LSTM model with the filtered data. We followed the same procedure to denoise the average speed and space headway data for all penetration rates of CVs.

(a) Filtered speed profile using standard Kalman filter
(b) Filtered space headway profile using standard Kalman filter
(c) Filtered speed profile using RTS filter
(d) Filtered space headway profile using RTS filter

Fig. 8. Comparison between average speed profile (and space headway profile) with 100% CV penetration rate and filtered speed profile (and space headway profile) with 10% CV penetration rate using standard Kalman Filter and RTS filters.
The performance of the traffic data prediction using LSTM in conjunction with different noise reduction models (the standard Kalman filter and RTS) was compared with the baseline model (i.e., LSTM without noise reduction model). Both the Mean Absolute Percentage Error (MAPE), RMSE and MAE are used as performance evaluation metrics. As the mathematical formulation for RMSE and MAE are given before, here we provide the mathematical formulation of MAPE:

$$\text{MAPE} = \frac{100\%}{N} \sum_{i=1}^{N} \frac{|y_i - \hat{y}_i|}{y_i}$$

where:
- $N$ represents the total sample size
- $y_i$ is the actual value of traffic data (speed or space headway) and
- $\hat{y}_i$ is the predicted value of traffic data (predicted speed or predicted space)

In Figures 9, 10, and 11, we compare the speed prediction for the RMSE, MAE and MAPE values, respectively, for the LSTM with the two noise reduction filters, a standard Kalman Filter and RTS, along with a typically used moving average data smoothing technique and LSTM without a noise reduction filter. All RMSE, MAE and MAPE values were calculated based on normalized ground truth and the predicted value of speed using the optimal hyperparameters for each of the prediction models as identified in Section 4.4. The range of the speed data normalization scale is from zero (0) to one (1). As shown in Figure 9, based on the comparison of the RMSE values, the results indicate that the LSTM combined with the three noise reduction models performs better than the LSTM alone. More specifically, the LSTM combined with the RTS filter provided lower RMSE values compared to filtered speed data using the Moving Average and the standard Kalman filter. For example, this RMSE value was reduced from 0.101 using LSTM only to 0.051 using the combination of LSTM and RTS filter at a 5% CVs penetration rate. Figure 10 shows a similar series of comparison results for the MAE values.

![Fig. 9. Comparison of RMSE for speed prediction between the LSTM with Moving Average, standard Kalman and RTS noise reduction filters, and LSTM without noise a reduction filter.](image-url)
As shown in Figure 11, compared to LSTM alone, LSTM combined with the RTS noise reduction model reduced MAPE from 19% to 5% for speed prediction at a 5% penetration of CVs. This research found that LSTM combined with the RTS noise reduction model reduced MAPE ranges from 1% to 14% for speed prediction from LSTM without using noise reduction filter for different penetration of CVs.
Figures 12, 13 and 14 compare the RMSE, MAE and MAPE values for space headway prediction, respectively, for the LSTM with the three noise reduction models explored here and the baseline model (LSTM without a noise reduction filter). Similar to the speed prediction, the results indicate that LSTM combined with all three noise reduction models performs better than the baseline model. Furthermore, LSTM combined with RTS filter provided the lowest MAE values similar to the RMSE values.

Fig. 12. Comparison of RMSE for space headway prediction between LSTM with Moving Average, standard Kalman and RTS noise reduction filters, and LSTM without noise reduction filter.

Fig. 13. Comparison of MAE for space headway prediction between LSTM with Moving Average, standard Kalman and RTS noise reduction filters, and LSTM without noise reduction filter.
Fig. 14. Comparison of MAPE for space headway prediction between LSTM with Moving Average, standard Kalman and RTS noise reduction filters, and LSTM without noise reduction filter.

As shown in Figure 15, compared to the baseline model, the LSTM combined with the RTS noise reduction model reduced the MAPE from 27% (using LSTM only) to 9% for a space headway prediction with a 5% penetration of CVs. The results indicated that the LSTM combined with the RTS noise reduction model reduced MAPE ranges from 2% to 18% for space headway prediction with different penetrations of CVs compared to the corresponding MAPEs for the baseline model.

Figures 15 and 16 compare the ground truth data (actual speed/space headway using 100% connected vehicles), and the predicted speed and space headway profile using the LSTM model combined with RTS models with 5% and 10% penetration rates of CVs (we used 5% and 10% penetration of CVs as examples). The predicted speed profiles of speed and space headway visually are also detailed in those figures.
Fig. 15. Comparison of ground truth and predicted speed data using LSTM combined with RTS for 5% and 10% penetration rates of CVs.

Fig. 16. Comparison of ground truth and predicted space headway data using LSTM combined with RTS for 5% and 10% penetration rates of CVs.
Table 1 summarizes the RMSE, MAE and MAPE values using LSTM combined with the RTS model for speed and space headway prediction. These values indicate superior performance of the developed prediction model performs with an increase in CV penetration. Table 2 summarizes the statistical significance test to identify significant differences of the predicted from the actual data. For this purpose, the t-test was conducted at a 95% confidence interval, the results of which indicated significant differences of the predicted speed using only LSTM is with the actual value for CV penetration ranging from 5% to 50%, and that of the space headway from the actual value for the 5% to 60% penetration range. No significant difference was observed between the predicted speed and space headway values using LSTM combined with RTS from the actual value from 5% to 100% CV penetration.
Table 1 Summary of RMSE, MAE and MAPE values using LSTM combined with RTS

| Traffic Data | Measure of Effectiveness | Penetration of Connected Vehicles |
|--------------|--------------------------|----------------------------------|
| Speed        |                          | 5%  | 10%  | 20%  | 30%  | 40%  | 50%  | 60%  | 70%  | 80%  | 90%  | 100% |
|              |                          |     |      |      |      |      |      |      |      |      |      |      |
| LSTM with RTS| RMSE                     | 0.035| 0.027| 0.025| 0.017| 0.018| 0.016| 0.018| 0.017| 0.015| 0.015| 0.013|
|              | MAE                      | 0.027| 0.023| 0.017| 0.016| 0.015| 0.013| 0.012| 0.012| 0.01| 0.01| 0.009|
|              | MAPE (%)                 | 4.99 | 4    | 3.1  | 3    | 2.9  | 2.57 | 2.49 | 2.45 | 2.42 | 2.4 | 2.09 |
| Space headway| RMSE                     | 0.051| 0.029| 0.028| 0.024| 0.028| 0.022| 0.024| 0.027| 0.02| 0.019| 0.017|
|              | MAE                      | 0.035| 0.021| 0.018| 0.016| 0.018| 0.013| 0.013| 0.013| 0.008| 0.008| 0.007|
|              | MAPE (%)                 | 9.02 | 6.62 | 5.6  | 5.6  | 5.1  | 4.07 | 4.36 | 3.2  | 2.48 | 2.49 | 2.4 |

Table 2 Summary of Statistical Significance Test

| Traffic Data | Penetration of Connected Vehicles |
|--------------|----------------------------------|
|              | 5%  | 10%  | 20%  | 30%  | 40%  | 50%  | 60%  | 70%  | 80%  | 90%  | 100% |
|              | Baseline model (LSTM without noise reduction filter) |
| Speed        | x   | x    | x    | x    | x    | x    | √    | √    | √    | √    | √    |
| Space headway| x   | x    | x    | x    | x    | x    | √    | √    | √    | √    | √    |
|              | LSTM with RTS                   |
| Speed        | √   | √    | √    | √    | √    | √    | √    | √    | √    | √    | √    |
| Space headway| √   | √    | √    | √    | √    | √    | √    | √    | √    | √    | √    |

Note: x = the actual and predicted values significantly different with 95% confidence interval; √ = the actual and predicted values are not significantly different at 95 % confidence interval.
6. **REAL-TIME APPLICATION EFFICACY**

The computational time required for a traffic data prediction model must be negligible so that the predicted traffic data can be used for route planning and scheduling, for future traffic condition assessment, and for vehicle’s energy optimization, all in real-time. In our analyses, individual vehicle-generated data were aggregated at time intervals of one-tenth of a second (100 milliseconds) to predict traffic speed and space headway. As the real-time application requires instant analysis of the aggregated data, LSTM model must be trained for a roadway corridor beforehand to have the computation time within the limit of the time requirements for any real-time traffic prediction (Rahman et al., 2018). Our analyses results indicate that the LSTM combined with the RTS model requires 80 milliseconds, on an average, to predict the speed and space headway, which is acceptable for real-time mobility and environmental applications (ARC-IT, 2018). We have used an Intel(R) Core(TM) i5-3210M CPU@2.5GHz and 6.00GB installed memory to run the LSTM/RTS prediction model.

![Diagram](image)

**Fig. 17. Field experiments of communication latency between RSU and connected vehicles.**

In a connected vehicle environment, the developed prediction model will be implemented in a roadside unit (RSU) that includes a data processing unit, such as an Intel® NUC device, which has a similar processing capability we have used in our experiments, and a Dedicated Short-Range Communication (DSRC) based roadside unit (RSU) that communicates with connected vehicles (as shown in Figure 17). We conducted a field experiment at the Clemson University-Connected and Autonomous Vehicle Testbed (CU-CAVT) to determine the two-way communication latency through DSRC between the CVs and a roadside unit for any real-time traffic application. In our field experiments on a roadway segment of Perimeter road at Clemson, South Carolina, we found that the two-way communication latency is 9 milliseconds on an average (as shown in Table 3). However, communication latency can vary because of the environmental inferences, such as trees, roadway slope, and curvature. Thus, we also determined the maximum (i.e., 22 milliseconds) and the minimum latency (i.e., 5 milliseconds) for two-way communication between an RSU and a CV as provided in Table 3. We found that total latency including computational time and two-way communication latency for our traffic data prediction application ranges from 85 milliseconds to
102 milliseconds. According to Southeast Michigan Test Bed Concept of Operations report, which is developed for the U.S. Department of Transportation (USDOT) to support connected vehicle research and development, the minimum latency for mobility and environmental applications should be within 1000 milliseconds (Fehr et al., 2014). As shown in Table 3, the total latency is much lower than the minimum latency requirement (approximately 1000 milliseconds), which would make our LSTM/RTS model suitable for real-time speed and space headway prediction.

Table 3 Summary of Two-way Communication Latency and Computation Time for the Real-time Traffic Data Prediction Application

| Two-way communication latency between CVs and RSU using DSRC | Computation time for running the prediction model | Total latency including two-way communication and computation time | Minimum Latency Requirements for Mobility and Environmental Application (Fehr et al., 2014) |
|------------------------------------------------------------|--------------------------------------------------|-----------------------------------------------------------------|-----------------------------------------------------------------|
| 22 ms (Maximum Latency)                                    | 80 ms                                            | 102 ms                                                          | ≤ 1000 ms                                                       |
| 9 ms (Average Latency)                                      | 80 ms                                            | 89 ms                                                           |                                                                 |
| 5 ms (Minimum Latency)                                      | 80 ms                                            | 85 ms                                                           |                                                                 |

7. CONCLUSIONS

This paper presents a real-time prediction model for traffic data that addresses the challenges of the stochastic nature of traffic flow over time and limited data due to a low CV penetration rate. The model combines a noise reduction model with an LSTM to reduce noise in the traffic data due to low penetration of CVs. We investigated noise reduction models, the standard Kalman filter and Kalman filter based RTS data smoothing techniques, to reduce the noise from the average speed and average space headway measured from BSMs and evaluated the performance of the LSTM prediction model for predicting traffic data using the resulting filtered data. The average speed and average space headway data used in this study were generated from the Enhanced NGSIM dataset, which contains vehicle trajectory data for every one-tenth of a second (similar to the broadcasting rate of BSMs in a CV environment).

We evaluated the prediction model to predict the average speed and average space headway using a CV penetration rate ranging from 5% to 100%. The analyses revealed that the model developed in this study can predict speed and space headway for different penetrations of connected vehicles with no significant difference from the ground truth data. Specifically, LSTM combined with an RTS noise reduction model reduced MAPE from 19% to 5% for speed prediction and from 27% to 9% for space headway prediction at a 5% penetration of CVs compared to the baseline model (LSTM without any noise reduction model). On the other hand, the reduction of MAPE value ranges from 1% to 14% for speed and 2% to 18% for space headway prediction with penetration rates of CVs ranging from 5% to 100% compared to the baseline model. A comparison of the standard Kalman filter and RTS filter along with the typically used moving average filter suggests that LSTM combined with RTS can achieve the best prediction performance in terms of RMSE, MAE and MAPE. The statistical significance test with a 95% confidence interval confirmed that the predicted speed and space headway using LSTM combined with RTS
is not significantly different from the ground truth for 5% to 100% CV penetration. In addition, the prediction accuracy of the average speed and space headway improves as the penetration of CV increases. The LSTM combined with the RTS model requires 80 milliseconds, on an average, to predict the traffic speed and space headway, which is within the limit for real-time traffic prediction in a connected vehicle environment.

The evaluation of our prediction model considered only peak hour freeway traffic data. Future studies will entail a further evaluation of this model developed here with the inclusion of i) both peak and off-peak hour traffic data in the model to evaluate the performance; ii) an arterial roadway scenario with signalized intersections; and iii) the implementation and evaluation of the model in a real roadway segment.

ACKNOWLEDGMENTS
This material is based on the study supported by the Center for Connected Multimodal Mobility (C\textsuperscript{2}M\textsuperscript{2}) (USDOT Tier 1 University Transportation Center) Grant headquartered at Clemson University, Clemson, South Carolina, USA. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Center for Connected Multimodal Mobility (C\textsuperscript{2}M\textsuperscript{2}), and the U.S. Government assumes no liability for the contents or use thereof.

REFERENCES
ARC-IT (Architecture Reference for Cooperative and Intelligent Transportation) (2018). Available at: https://local.iteris.com/arc-it/index.html. Accessed January 2018.

Bergstra, J., and Bengio, Y. (2012). Random search for hyper-parameter optimization. Journal of Machine Learning Research 13, no. Feb (2012): 281-305.

Chowdhury, M., Sadek, A., Ma, Y., Kanhere, Y., N., and Bhavsar, P. (2006). Applications of Artificial Intelligence Paradigms to Decision Support in Real-time Traffic Management. Transportation Research Record, Journal of Transportation Research Board, Vol. 1968(1), 92-98.

Du, Y., Chowdhury, M., Rahman, M., Dey, K., Apon, A., Luckow, A., and Ngo, L. B. (2017). A Distributed Message Delivery Infrastructure for Connected Vehicle Technology Applications. IEEE Transactions on Intelligent Transportation Systems, 19(3), 787-801.

Ervin, R.D., MacAdam, C.C., Gilbert, K., and Tchoryk, P. (1991). Quantitative Characterization of the Vehicle Motion Environment (VME). In Proceedings of the 2nd Vehicle Navigation and Information Systems Conference, 20–23 October 1991, Dearborn, Mich. Society of Automotive Engineers, Inc., Warrendale, Penn. pp. 1011 - 1029. doi:10.1109/VNIS.1991.205846.

Fehr, W. (2018). Southeast Michigan Test Bed: 2014 Concept of Operations. https://local.iteris.com/cvria/docs/SE_Michigan_Test_Bed_2014_ConOps_-_v1_-_2014-Dec_29.pdf, Accessed Feb 14, 2018.
Gadsden, S. A. and Lee, A. S. (2017). Advances of the Smooth Variable Structure Filter: Square-Root and Two-Pass Formulations. *Journal of Applied Remote Sensing*, 11(1), 015018.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.*, 9(8), 1735–1780.

Hua, Y., Zhao, Z., Li, R., Chen, X., Liu, Z., and Zhang, H. (2017). Traffic Prediction Based on Random Connectivity in Deep Learning with Long Short-Term Memory. *arXiv preprint, arXiv:1711.02833*.

Ishak, S., Kotha, P. and Aleksandru, C. (2003). Optimization of Dynamic Neural Network Performance for Short-Term Traffic Prediction. *Transport. Transportation Research Record: Journal of Transportation Research Board*, 1836, 45–56.

Kingma D. P. and J. Ba. (2014). Adam: A Method for Stochastic Optimization. [Online]. Available: https://arxiv.org/abs/1412.6980.

Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1), 35-45.

Khan, S. M., Dey, K. C., and Chowdhury, M. (2017). Real-Time Traffic State Estimation with Connected Vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 18(7), 1687-1699.

Kanagaraj, V., Asaithambi, G., Toledo, T., and Lee, T.C. (2015). Trajectory Data and Flow Characteristics of Mixed Traffic. *Transportation Research Record, Journal of Transportation Research Board*, 2491: 1-11. doi:10.3141/2491-01.

Liu, J. and Khattak, A.J., (2016). Delivering Improved Alerts, Warnings, and Control Assistance Using Basic Safety Messages Transmitted Between Connected Vehicles. *Transportation research part C: Emerging Technologies*, 68, pp.83-100.

Montanino, M. and Punzo, V., 2013. Making NGSIM data usable for studies on traffic flow theory: Multistep method for vehicle trajectory reconstruction. Transportation Research Record, 2390(1), pp.99-111.

Ma, X., Tao, Z., Wang, Y., Yu, H., and Wang, Y. (2015). Long Short-Term Memory Neural Network for Traffic Speed Prediction Using Remote Microwave Sensor Data. *Transportation Research Part C: Emerging Technologies*, 54, 187-197.

Ma, Y., Chowdhury, M., Sadek, A., and Jeihani, M. (2009). Real-Time Highway Traffic Condition Assessment Framework using Vehicle–Infrastructure Integration (VII) with Artificial Intelligence (AI). *IEEE Transactions on Intelligent Transportation Systems*, 10(4), 615-627.
Ma, Y., Chowdhury, M., Sadek, A., and Jeihani, M. (2012). Integrated Traffic and Communication Performance Evaluation of an Intelligent Vehicle Infrastructure Integration (VII) System for online Travel-Time Prediction. *IEEE Transactions on Intelligent Transportation Systems*, 13(3), 1369-1382.

NGSIM. (2006). NGSIM I-80 Data Analysis. Cambridge Systematics, Inc., Cambridge, MA, USA, Summary report, 2006.

Ossen, S., and Hoogendoorn, S.P. (2008). Validity of Trajectory-Based Calibration Approach of Car-Following Models in Presence of Measurement Errors. *Transportation Research Record*, 2088: 117-125. doi:10.3141/2088-13.

Park, D. and Rilett, L.R., (1999). Forecasting Freeway Link Travel Times with a Multilayer Feedforward Neural Network. *Comput. Civ. Infrastruct. Eng.* 14 (5), 357–367.

Punzo, V., Formisano, D., and Torrieri, V. (2005). Part 1: Traffic Flow Theory and Car-following: Nonstationary Kalman Filter for Estimation of Accurate and Consistent Car-Following Data. *Transportation Research Record*, 1934: 1-12. doi:10.3141/1934-01.

Punzo, V., Borzacchiello, M.T., and Ciuffo, B. (2011). On the Assessment of Vehicle Trajectory Data Accuracy and Application to the Next Generation Simulation (NGSIM) Program Data. *Transportation Research Part C: Emerging Technologies*, 19: 1243 – 1262. doi:10.1016/j.trc.2010.12.007.

Rahman, M., Chowdhury, M., Rayamajhi, A., Dey, K., and Martin, J. (2018). Adaptive Queue Prediction Algorithm using Edge-Centric Cyber–Physical System Platform in a Connected Vehicle Environment. In the Proceeding of 97th Annual Meeting of the Transportation Research Board, Washington, D.C.

Rauch, H. E., Striebel, C. T., and Tung, F. (1965). Maximum Likelihood Estimates of Linear Dynamic Systems. *AIAA journal*, 3(8), 1445-1450.

Ren, H., Song, Y., Liu, J., Hu, Y., and Lei, J. (2017). A Deep Learning Approach to the Prediction of Short-term Traffic Accident Risk. arXiv preprint arXiv:1710.09543.

Rim, H., Park, S., Oh, C., Park, J., and Lee, G. (2016). Application of Locally Weighted Regression-Based Approach in Correcting Erroneous Individual Vehicle Speed Data. *Journal of Advanced Transportation*, 50(2): 180–196. doi:10.1002/atv.1325.

Thiemann, C., Treiber, M., and Kesting, A. (2008). Estimating Acceleration and Lane-changing Dynamics from Next Generation Simulation Trajectory Data. *Transportation Research Record*, 2088: 90-101. doi:10.3141/2088-10.

Toledo, T., Koutsopoulos, H. and Ahmed, K., 2007. Estimation of vehicle trajectories with locally weighted regression. *Transportation Research Record: Journal of the Transportation Research Board*, (1999), pp.161-169.
Vanajakshi, L., and Rilett, L. R. (2004). A comparison of the Performance of Artificial Neural Networks and Support Vector Machines for the Prediction of Traffic Speed. In *Intelligent Vehicles Symposium*, 2004 IEEE, 194-199.

Van Lint, J.W.C., Hoogendoorn, S.P., Zuylen, H.V. (2002). Freeway Travel Time Prediction with State-spaced Neural Networks: Modeling State-space Dynamics with Recurrent Neural Networks. *Transportation Research Record Journal: Transportation Research Board* 1811, 30–39.

Van Lint, J.W.C., Hoogendoorn, S.P., and Zuylen, H.V., (2005). Accurate Freeway Travel Time Prediction with State-space Neural Networks under Missing Data. *Transportation Research Part C* 13, 347–369.

Van Zuylen, H.J. and Muller, T.H., (2002). Regiolab delft. In 9th World Congress on Intelligent Transport SystemsITS America, ITS Japan, ERTICO (Intelligent Transport Systems and Services-Europe).

Vanajakshi, L. and Rilett, L. R., 2004. A Comparison of the Performance of Artificial Neural Networks and Support Vector Machines for the Prediction of Traffic Speed. *IEEE Intelligent Vehicles Symposium*, pp. 194-199.

Wang, X., Jiang, R., Li, L., Lin, Y., Zheng, X. and Wang, F.Y., (2018). Capturing Car-following Behaviors by Deep Learning. *IEEE Transactions on Intelligent Transportation Systems*, 19(3), pp.910-920.

Zhao, Z., Chen, W., Wu, X., Chen, P. C., and Liu, J. (2017). LSTM Network: A Deep Learning Approach for Short-term Traffic Forecast. *IET Intelligent Transport Systems*, 11(2), 68-75.