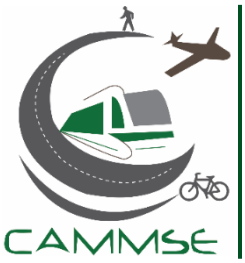


Center for Advanced Multimodal Mobility Solutions and Education

Project ID: 2022 Project 15

PREDICTION OF TRAFFIC MOBILITY BASED ON HISTORICAL DATA AND MACHINE LEARNING APPROACHES

Final Report

by

Yong Deng (ORCID ID: https://orcid.org/0000-0002-4299-1936)
Chuang Chen (ORCID ID: https://orcid.org/0000-0002-1239-1833)
Xianming Shi (ORCID ID: https://orcid.org/0000-0003-3576-8952)

Department of Civil and Environmental Engineering
PO Box 642910, Pullman, WA, 99164.
Phone: 1-509-335-7088; Email: Xianming.Shi@wsu.edu

for

Center for Advanced Multimodal Mobility Solutions and Education
(CAMMSE @ UNC Charlotte)
The University of North Carolina at Charlotte
9201 University City Blvd
Charlotte, NC 28223

August 2022

ACKNOWLEDGEMENTS

This project was funded by the Center for Advanced Multimodal Mobility Solutions and Education (CAMMSE @ UNC Charlotte), one of the Tier 1 University Transportation Centers that were selected in this nationwide competition, by the Office of the Assistant Secretary for Research and Technology (OST-R), U.S. Department of Transportation (US DOT), under the FAST Act. The authors are also very grateful for all of the time and effort spent by the transportation professionals to provide the data that was critical for the successful completion of this study.

DISCLAIMER

Table of Contents

List of Figures

List of Tables

EXCUTIVE SUMMARY

Traffic mobility plays an important role in the intelligent transportation system (ITS). Traffic mobility is a factor significantly affecting road safety and efficiency (as well as environmental stewardship), and its prediction has attracted continuous attention over the past decades. With the rapid development of machine learning (ML) techniques, the accuracy and stability of predictive models for traffic mobility have been improved substantially.

The goal of this project is to develop predictive models for traffic mobility using ML approaches. The focus is placed on two essential components - traffic speed and traffic volume. To this end, this project addresses the following objectives: (1) identifying appropriate WSDOT highway segments for this modeling study and collecting the relevant historical data related to traffic mobility; (2) developing ML models suitable for predicting the traffic mobility, from model type selection to model validation; (3) comparing different ML models and traditional models in terms of accuracy and stability; and (4) selecting desirable models according to their prediction performance for future studies.

Traffic speed and traffic volume of the Interstate highway I-5 were modelled in this project. A total of 8928 dataset was collected for both speed and volume at each milepost along this highway throughout 2016 with the interval of five minutes. The data at a randomly selected milepost was further compressed to 744 with the length of one month and interval of one hour. Out of the 744 records, the first 672 (28 days x 24 hours) were utilized for model training and the rest 72 (3 days x 24 hours) were utilized for model validation.

This work explored the use of the statistical model seasonal autoregressive integrated moving average (SARIMA), traditional ML model multilayer perceptron (MLP) and deep learning (DL) models convolutional neural network (CNN) and long short-term memory (LSTM), in the modeling and prediction of traffic speed and traffic volume. For each of these models, mathematical expression and illustrative model structure were provided; and model construction procedures from hyperparameter determination, input and output process, parameter calibration to model performance evaluation and comparison were introduced. Specifically, typical indicators were utilized for model fitting and prediction performance evaluation and comparison, such as coefficient of determination (R^2) and root mean square error (RMSE) for model accuracy, autocorrelation function (ACF) and partial autocorrelation function (PACF) values for model residual correlations, confidence interval for model stability and Shapiro-Wilk statistic for residual normality.

In general, SARIMA, MLP, CNN and LSTM achieved similar and satisfactory performance in both fitting and predicting traffic volume development. In contrast, the performance of these models was poorer for traffic speed, possibly due to its data characteristics such as distribution and dependencies.

Chapter 1. Introduction

1.1 Problem Statement

Mobility refers to the movement of people or goods (Litman, 2003). Therefore, from a traffic mobility perspective, capacity and speed of vehicle system are two major factors to be focused on and improved (Litman, 2003). Accurate prediction of traffic mobility leads to safer driving, better road user experience, cleaner environment, and better utilization and planning of infrastructures. It also contributes to a clearer understanding of traffic mobility interactions with external factors such as climatic conditions and maintenance actions (Chen & Shi, 2019). Furthermore, benefits in economic and social dimensions from traffic mobility prediction have been revealed from cost-benefit analysis (CBA) (Peer, Koopmans, & Verhoef, 2012) and social impact assessment (SIA) (Anciaes, Metcalfe, & Heywood, 2017). The difficulties of developing reliable predictive models for traffic mobility, lie in the dynamic dependencies on space and time (Yin et al., 2021) as well as high reliance on external factors such as on weather conditions, events, road attributes and maintenance strategies.

Due to the complexity of traffic system and various influencing factors as mentioned above, traditional model-based methods such as kinematic wave often fail to provide timely and accurate predictions for real-time traffic mobility (Zhou et al., 2022). On the other hand, the focus is increasingly shifted to the data-driven models to capture the pattern of traffic mobility components and their relationships with external factors. With the fast development in artificial intelligence (AI), spatial, temporal, and external factors dependencies can be captured by either the complex model structures or learning algorithms of machine learning (ML) in particular, deep learning (DL) models. Meanwhile, with the increasing computing power and data mining techniques, ML models have become a useful and accessible tool to predict traffic mobility.

1.2 Objectives

The goal of this project is to develop predictive models for traffic mobility using ML approaches. The focus is placed on two essential components - traffic speed and traffic volume. To this end, this project addresses the following objectives: (1) identifying appropriate WSDOT highway segments for this modeling study and collecting the relevant historical data related to traffic mobility; (2) developing ML models suitable for predicting the traffic mobility, from model type selection to model validation; (3) comparing different ML models and traditional models in terms of accuracy and stability; and (4) selecting desirable models according to their prediction performance for future studies.

This scope is directly relevant to the CAMMSE theme of "*Developing data modeling and analytical tools to optimize passenger and freight movements*". The research results will allow state departments of transportation (DOTs) and other roadway agencies to achieve better understanding of ML models and basic procedures to predict traffic mobility from its historical records. The developed models and model construction techniques may contribute to future studies such as investigations of external factor effects on traffic mobility and assessments of the economic and social benefits. This research will ultimately translate to better decision-making and management practices in the traffic management system and pavement management system.

1.3 Expected Contributions

To accomplish these objectives, several tasks have been undertaken.

Task 1. Literature review on ML models for traffic mobility. The WSU team conducted a thorough and targeted review on the literature of traffic mobility prediction using ML approaches. Databases including TRID, Google Scholar, ISI Web of Science, and IEEE Xplore were searched and pertinent published literature (e.g., scholarly journal articles, proceedings, and technical reports) covered topics such as surveys on ML models for the traffic mobility prediction and their comparisons with each other and with traditional models.

Task 2. Data collection and processing of traffic mobility. The WSU team selected the highway segments of interest, considering data availability, data quality, etc. This was based on the hands-on experience gained in the previous CAMMSE project titled "Modeling the macroscopic effects of winter maintenance operations on traffic mobility on Washington highways" (Chen & Shi, 2019). For example, interstate highways I-5 and I-90 were continued to be studied with extended historical records. Characteristics of traffic mobility such as the traffic volume and speed were collected by loop detectors in the State of Washington and are accessible via UWDRIVE (http://uwdrive.net/STARLab). The observational period is five minutes for all lanes, which provides the team adequate and valuable information on the traffic pattern.

Task 3. Modeling of traffic mobility using ML. Historical data of traffic mobility collected from Task 2 were processed for the predictive model development, summarized in the following procedure. The first step is to split the collected data into the training dataset and the validation dataset. Then, the data were examined and processed to satisfy the assumptions or requirements of applied models. For example, seasonality of data was examined to determine whether ARIMA or seasonal-ARIMA (SARIMA) should be applied in the modelling. Following that, different traditional and ML models were trained using the same training dataset and predicted a subsequential series as the validation dataset. Next, a comprehensive comparison between different models was conducted with collected and predicted validation datasets. Critical indicators for the model performance evaluation such as root mean square error (RMSE) and residual plot were utilized. Finally, the recommended model was provided according to their prediction performance.

Task 4. Final report and technology transfer. This final report was written and submitted to CAMMSE UTC and a technical paper will be prepared and submitted to a journal for peer review and publication.

1.4 Report Overview

The remainder of this report is organized as follows: Chapter 2 presents a brief review of the state-of-the-art literature on predictive models of traffic mobility using ML approaches. Chapter 3 describes the methodology used in this study, including the data collection and processing methods and introduction of statistical, traditional ML and DL models applied in this study. Chapter 4 describes the relative prediction performance of the applied models. Finally, Chapter 5 concludes this report with conclusions and a discussion of the directions for future research.

Chapter 2. Literature Review

2.1 Introduction

Figure 2.1 shows typical model types for vehicle speed prediction. Those models can be utilized for traffic mobility prediction as well and are introduced in this chapter. It can be seen that traditional ML and DL models occupy a large proportion in data-driven methods, reflecting the trend of applying AI in traffic mobility prediction and its desirable performance. AI refers to techniques in which computers mimic human behaviors to solve complex tasks with or without human intervention (Russell & Norvig, 2021). As a subfield of AI, ML is the process of a computer learning information from mathematical models of data. The computer system iteratively learns and improves from the problem-specific data without the hidden insights and patterns explicitly programmed (Janiesch, Zschech, & Heinrich, 2021). As a subfield of ML, DL applied model structures and operations of higher complexity and advances in contrast to traditional ML. It shows superior performance in processing large and high-dimensional data.

According to a recent survey conducted on cutting-edge predictive models for traffic (Do, Taherifar, & Vu, 2019), over 85 percent of models predicting the short-term state of traffic are neural networks, of which the traditional ML models include the most applied artificial neural network (ANN), and the DL models include deep neural network (DNN), convolutional neural network (CNN) and recurrent neural network (RNN), etc. as shown in Figure 2.1. These models are distinguished with their structures and operations as shown in Figure 2.2. This section focuses on representative classical data-driven models and DL models applied in predicting traffic mobility.

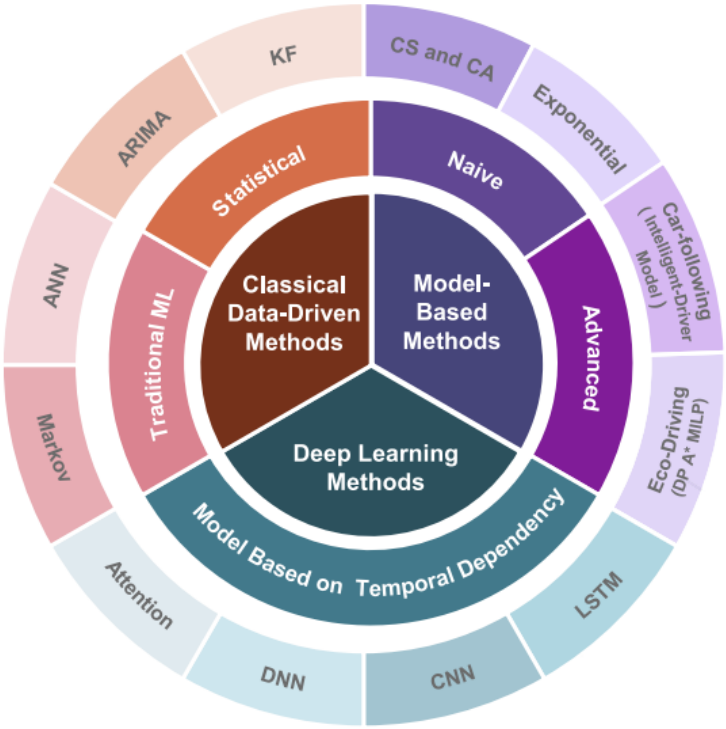


Figure 2.1 Model Types for Vehicle Speed Prediction (Zhou et al., 2022)

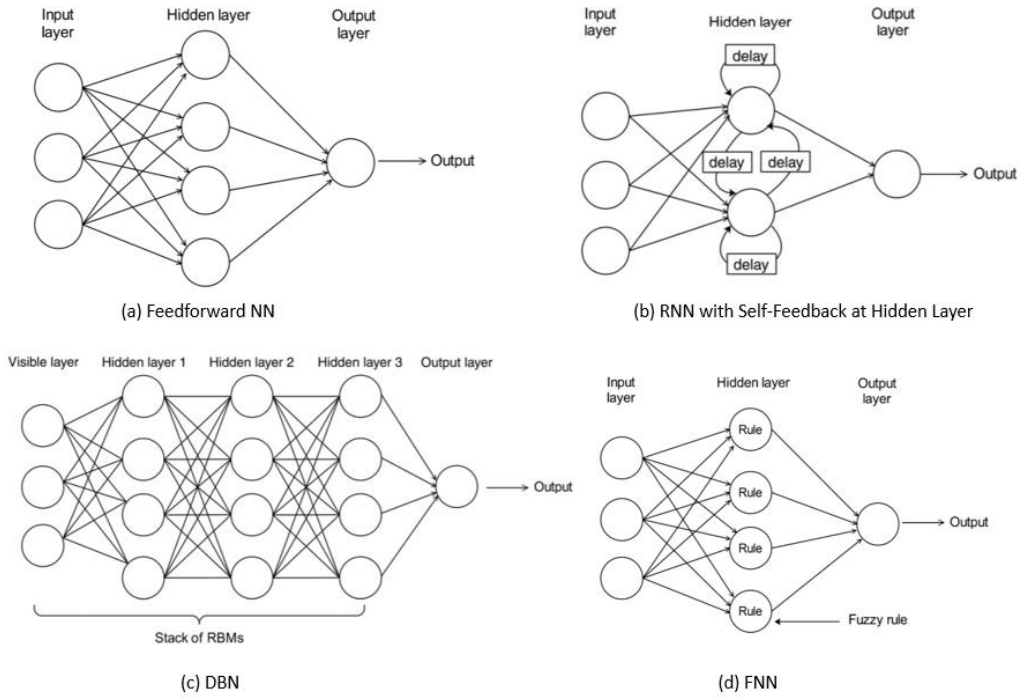


Figure 2.2 Structures of Typical ANN Models (Do et al., 2019)

2.2 Statistical Models

Autoregressive integrated moving average (ARIMA) model is widely applied in modelling time series. It combines the autoregressive (AR) model and moving average (MA) model (Box & Jenkins, 1970). The weighted historical data and prediction errors are considered in the model form. In addition to stationary data, time series with trend or seasonality can also be modelled with its extensions, e.g., SARIMA and vector form of ARIMA (VARIMA). Van Der Voort er al. (Van Der Voort, Dougherty, & Watson, 1996) combined Kohonen self-organizing map (Kohonen, 1990) as a data classifier and ARIMA model to improve the prediction performance of single ARIMA model for the short-term traffic forecasting. Williams and Hoel (Williams & Hoel, 2003) applied SRIMA to predict traffic flow (number of vehicles per hour). It outperformed heuristic forecast benchmarks such as random walk and historical average in two representative freeways. Chandra and Al-Deek (S. R. Chandra & Al-Deek, 2009) considered spatial effect on the traffic and applied VARIMA model to predict traffic speed and traffic volume at both upstream and downstream locations. Statistical methods such as crosscorrelation analysis were utilized to indicate the necessity of incomplete specifications and multivariate models in predicting spatial time series as traffic (S. R. Chandra & Al-Deek, 2009). Min and Wynter (Min & Wynter, 2011) proposed a multivariate spatial-temporal autoregressive (MSTAR) model which is refined from VARIMA model and considers spatial characteristics of road network. It achieved good accuracy in predicting traffic volume and speed.

The Kalman filter (KF) is an online learning algorithm proposed for the discrete linear filtering problem (Kalman, 1960). The estimates of the state variables and their uncertainties are made and then updated with observations for the next timeframe. Dailey (Dailey, 1999) proposed extended KF (EKF) for nonlinear systems by linearizing all nonlinear models. It was applied to predict traffic speed using traffic volume and occupancy data in the study. Julier and Uhlmann

(Julier & Uhlmann, 1997) proposed unscented KF (UKF) which uses weighted points to parameterize the means and covariances of probability distribution. It has the higher accuracy and less implementation difficulty than EKF for nonlinear systems. KF and its extensions and generalizations have gained wide applications in predicting traffic speed and volume (Guo, Xia, & Smith, 2009; Y. Wang & Papageorgiou, 2005; Y. Wang, Papageorgiou, & Messmer, 2008; Ye, Zhang, & Middleton, 2006).

2.3 Traditional ML Models

Artificial neuron network (ANN) is inspired by the biological nervous system in which the information reception, conversion and transmission are simulated similar to how the brain neurons work (Deng & Shi, 2022a). In traditional ML models, it specifically refers to the shallow NN or multilayer perceptron (MLP) which consists of one input layer containing model inputs, one hidden layer containing hidden neurons and one output layer containing model outputs. The neuron weight and bias are calibrated with the training data via the backpropagation algorithm (Rumelhart, Hinton, & Williams, 1985). Although it is the simplest ANN, the relationship between model inputs and outputs can be captured by the neuron connection and activation function. Goudarzi (Goudarzi, 2018) compared ANN model with nearest neighbors algorithms and linear regression models in the travel time prediction. A shallow ANN was found to have higher accuracy than other models in predicting traffic time for short horizons. Besides, shallow NN was typically utilized to predict short-term traffic speed and volume in previous studies (Kumar, Parida, & Katiyar, 2013; Kumar, Parida, & Katiyar, 2015; Sharma, Kumar, Tiwari, Yadav, & Nezhurina, 2018). Xiao et al. (Xiao, Sun, & Ran, 2004) combined fuzzy logic and NN (FNN) to predict the traffic speed and quantify effects of special factors. The fuzzy rules were applied in the hidden layer for the output to show the impact of special factors (Xiao et al., 2004).

In graphical models or probabilistic graphical models (PGMs), conditional independence relationships between interacting variables are expressed by graphs (Gorinova, Gordon, Sutton, & Vákár, 2021). There are two types of PGMs - directed graphical models (DGMs) and undirected graphical models (UGMs) which are represented by Bayesian networks (BNs) and Markov random fields (MRFs), respectively. In DGMs, the directed edge can indicate the causal relationship of one variable to another, or conditional distribution in the perspective of probability. Accordingly, spatial and temporal dependencies of traffic state can be represented in DGMs. BNs have been utilized to predict traffic flow (Sun, Zhang, & Yu, 2006; Zhu, Peng, Xiong, & Zhang, 2016) and Hidden Markov Models (HMMs), as another subset of DGMs, have been utilized to predict traffic speed (Qi & Ishak, 2014; Rapant, Slaninová, Martinovič, & Martinovič, 2016).

2.4 DL Models

Compared with traditional ML models, model structures and algorithms of higher complexity and sophistication are applied in DL models. They were proposed not only to improve the performance of traditional ML models but also to capture the complicated nature of traffic characteristics and development. Deep belief network (DBN) (Hinton, Osindero, & Teh, 2006) stacks multiple restricted Boltzmann machines (RBMs). RBM belongs to UGMs and consists of a visible layer and a hidden layer. RBMs in the DBN are trained layer by layer and the hidden layer output of the previous RGB serves as the input for the next one (Bao, Jiang,

Yang, & Wang, 2021). DBN can be utilized for both unsupervised and supervised learning tasks including regression. It was applied to predict traffic flow in previous studies (Bao et al., 2021; Huang, Song, Hong, & Xie, 2014; Li et al., 2019; Y. Zhang & Huang, 2018).

Convolutional neural networks (CNNs) (LeCun & Bengio, 1995) were designed for images and speeches with convolutional filters to extract features. Those feature maps are then processed by the activation function, pooling and flattening layers and then serve as inputs of a fully connected layer for classification or regression. Components of traffic mobility can be represented by either time series (W. Zhang, Yu, Qi, Shu, & Wang, 2019) or images (Ma et al., 2017). Both data types can be processed by CNNs for prediction. Recently, traditional CNN was modified to improve the prediction performance and accurately captured the spatial and temporal dependencies of traffic mobility. Capsule network (CapsNet) replaces max pooling with capsules to reduce information loss in the pooling layer of CNN (Kim, Wang, Zhu, & Mihaylova, 2018). It achieved higher accuracy than traditional CNN in traffic speed prediction. Graph convolutional network (GCN) can deal with non-Euclidean data which include traffic of urban road network with complex topological structure (L. Zhao et al., 2019). The spatial features are captured by a filter in the frequency domain in GCN models.

Recurrent neural networks (RNNs) treat output of the current step as the input for the next one. Accordingly, they have advantages of capturing the temporal features of sequential data such as time series and text. To avoid the gradient exploding and vanishing occurred in traditional RNNs, long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997) with input, output and forget gates and gated recurrent units (GRUs) (Cho, Van Merriënboer, Bahdanau, & Bengio, 2014) with reset and update gates were proposed to improve the performance of traditional RNNs. Information to be retained for the next step can be determined by those gates. They have been applied in predicting traffic flow (Fu, Zhang, & Li, 2016; Z. Zhao, Chen, Wu, Chen, & Liu, 2017). Besides, they were coupled with CNN, GCN, etc. to capture both the spatial and temporal dependencies of traffic for better prediction performance and representativeness (Cao, Li, & Chan, 2020; X. Wang et al., 2018).

Attention mechanism in ML is a reflection of cognition attention. Important parts of the input are focused and the others are diminished (Niu, Zhong, & Yu, 2021). In RNN-based encoder-decoder models, the attention mechanism generates a vector for the relationship between encoder and decoder vectors. It helped to preserve useful information from the long past for LSTM units in the traffic flow prediction (Z. Wang, Su, & Ding, 2020). Besides, multi-headed attention layers were utilized in the transformer model to extract the spatial relationships between different grids in terms of traffic flow (Liu, Li, & Lu, 2021). Graph attention network (GAT) applies a self-attention mechanism to aggregate information of different neighbors using the attention score (Wu et al., 2020). It was used in the traffic flow and speed predictions in previous studies (Pan et al., 2019).

2.5 Summary

This chapter briefly reviews the predictive models for traffic mobility in terms of traffic speed and volume/flow. The focused model types are classic data-driven models including statistical and traditional ML models and DL models including CNN, LSTM, etc. The major findings in this chapter can be summarized as follows.

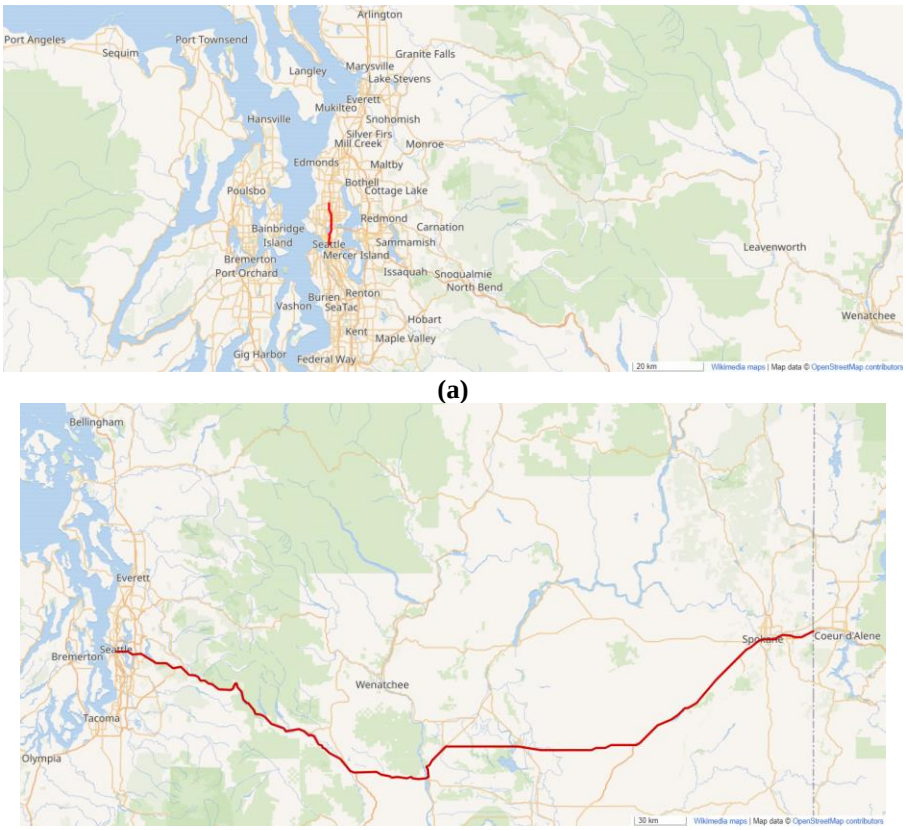
- Compared with classic data-driven models, DL models are developed with more advantages of capturing spatio-temporal dependencies of traffic mobility. They are reflected on the model structure, algorithm, etc.
- One should consider the characteristics of data in the selection of predictive model. Different models have corresponding applications in traffic mobility of single lane, multiple lanes, or road network.
- The developing trend of predictive models is the hybrid model that takes advantages of individual model modes to improve model accuracy, stability, robustness, etc.

Chapter 3. Methodology

This chapter includes two major parts - data collection and processing and introduction of applied predictive models. Section 3.1 describes the data source, type and processing method for the predictive model construction. Section 3.2 introduces representative statistical models - ARIMA and its extension. They serve as reference models in this project for the performance comparison with ML and DL models. Section 3.3 introduces a representative traditional ML model - MLP. Section 3.4 introduces two representative DL model - CNN and LSTM. Compared with their reviews in Chapter 2, the mathematical form and/or illustrative figure of model structure and construction procedures are detailed in this chapter.

3.1 Data Collection and Processing

As in a previous project (Chen & Shi, 2019), traffic volume and speed data were obtained from the installed detectors loops and traffic monitoring systems in Washington State. Loop detector data was collected in every 5 minutes and is recorded in UWDRIVE (http://uwdrive.net/STARLab). A total of 8928 dataset was collected for each milepost along two highways Interstate 5 (I-5) and Interstate 90 (I-90) throughout 2016. Figure 3.1 illustrates the locations of these two highways.

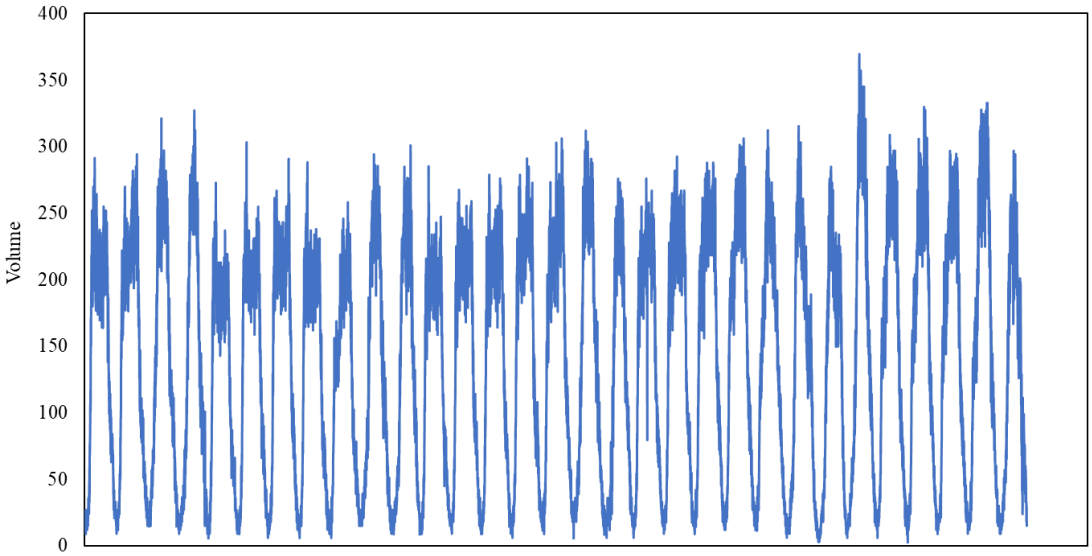


(a)

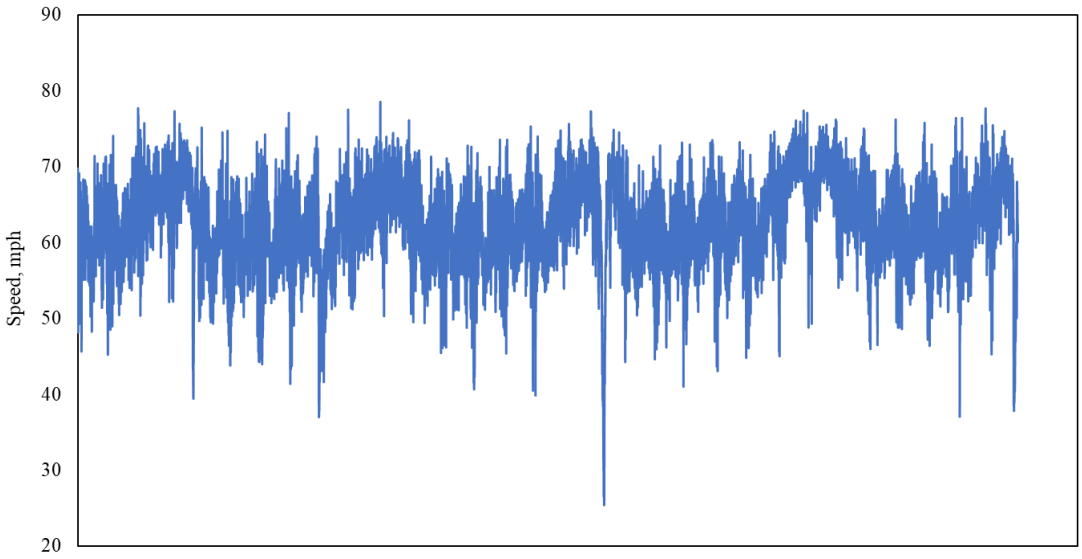
(b)

Figure 3.1: Locations of (a) I-5 and (b) I-90 (adapted from Google Map)

This project utilized data of a randomly selection milepost, the milepost "209.33" of I-5, as an example. Figure 3.2 presents the 5-minute traffic volume and speed data in December 2016. For the convenience of model construction and future analysis, the 5-minute data was aggregated into the hourly one as Figure 3.3, in which traffic volume was summed and traffic speed was averaged respectively. A total of 744 data (31 days x 24 hours) was organized for traffic volume and speed, in which the first 672 (28 days x 24 hours) was utilized for model training and the rest 72 (3 days x 24 hours) was utilized for model validation. Since this project focused on traffic mobility of an individual lane and no network was involved, data was treated as time series and corresponding statistical, ML and DL models were applied to capture its development pattern.



(a)



(b)

Figure 3.2: Collected data of (a) traffic volume and (b) traffic speed

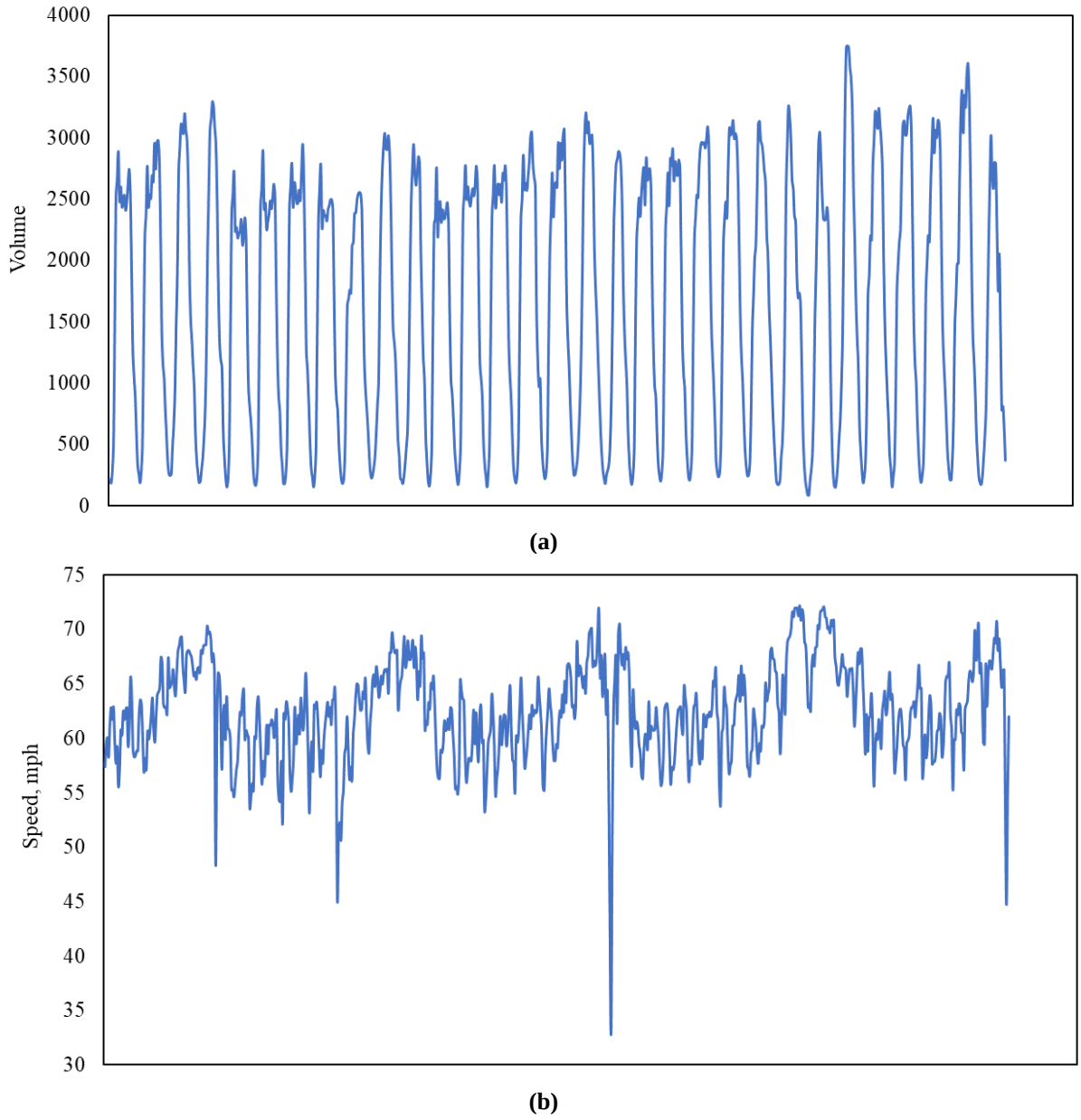


Figure 3.3: Processed data of (a) traffic volume and (b) traffic speed

3.2 ARIMA and SARIMA

ARIMA stands for autoregressive integrated moving average. Autoregression (AR) model describes the predicted variable using its history, constant and noise as Equation 1,

$$y_t = c + \sum_{i=1}^{p} \varphi_i y_{t-i} + \varepsilon_t \quad (1)$$

where,

y_t = predicted variable y at time t;

c = constant;

φ = model parameter;
ε = white noise (zero mean and constant variance).

Equation 1 shows that the predicted variable is the linear combination of its p lagged values. According, it is called an AR model of order *p* (*AR(p)*) (Hyndman & Athanasopoulos, 2018).

Moving average (MA) model describes the predicted variable as the linear combination of past prediction errors as Equation 2,

$$y_t = \mu + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t \tag{2}$$

where,
μ = mean of the series;
θ = model parameter.

Similar to AR model, Equation 2 can be called a MA model of order *q* (*MA(q)*). Finally, ARIMA model can be expressed as Equation 3,

$$y'_t = c + \sum_{i=1}^{p} \varphi_i y'_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t \tag{3}$$

where,
y'_t = differenced series.

Generally, ARIMA model with the AR order *p*, differencing order *d* and MA order *q* (*ARIMA(p,d,q)*) can be expressed as Equation 4 with the backshift notation *B* (Hyndman & Athanasopoulos, 2018).

$$\begin{aligned}
& y_t^{(d)} = c + \sum_{i=1}^{p} \varphi_i y_{t-i}^{(d)} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t \\
& \rightarrow \left(1 - \varphi_1 B - ... - \varphi_p B^p\right)\left(1-B\right)^d y_t = c + \left(1 + \theta_1 B + ... + \theta_q B^q\right)\varepsilon_t \\
& \rightarrow \left(1 - \varphi_1 B - ... - \varphi_p B^p\right)\left[\left(1-B\right)^d y_t - \mu\right] = \left(1 + \theta_1 B + ... + \theta_q B^q\right)\varepsilon_t \\
& \rightarrow \phi(B)\left[\left(1-B\right)^d y_t - \mu\right] = \theta(B)\varepsilon_t
\end{aligned} \tag{4}$$

in which the constant term μ can be expressed as Equation 5.

$$\mu = c \Big/ \left(1 - \sum_{i=1}^{p} \varphi_i\right) \tag{5}$$

Differenced series indicates that Equation 3 and Equation 4 are suitable for stationary (or stochastic) data in which the trend and seasonality are eliminated. Trend can be eliminated by the differencing and additional terms can be added to the ARIMA model for the seasonality. Seasonal ARIMA (SARIMA) model is typically denoted as $ARIMA(p,d,q)(P,D,Q)_m$ with *P*, *D*, *Q* and *m* representing the seasonal AR order, seasonal differencing, seasonal MA order and time

span of repeating seasonal pattern, respectively. Similar to Equation 4, SARIMA model can be expressed as Equation 6,

$$\Phi\left(B^m\right)\phi\left(B\right)\left[\left(1-B\right)^d\left(1-B^m\right)^D y_t-\mu'\right]=\Theta\left(B^m\right)\theta\left(B\right)\varepsilon_t \tag{6}$$

in which,

$$\begin{aligned}\Phi\left(B^m\right)&=1-\sum_{i=1}^{P}\Phi_i B^{mP}\\ \Theta\left(B^m\right)&=1+\sum_{i=1}^{Q}\Theta_i B^{mQ}\end{aligned} \tag{7}$$

Basic procedures of determining final ARIMA model for data fitting and prediction can be summarized as conducting stationarity test using Augmented Dickey Fuller (ADF) test (Dickey & Fuller, 1979) and/or Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test (Kwiatkowski, Phillips, Schmidt, & Shin, 1992), selecting potential nonseasonal and seasonal AR, MA and differencing orders using autocorrelation function (ACF) and partial autocorrelation function (PACF), and determining the final model with the optimal Akaike Information Criterion (AIC) (Akaike, 1974), Bayesian Information Criterion (BIC) (Schwarz, 1978), etc. Specifically, ACF and PACF provide correlations between two variables in time series. For data in this project, the time span of repeating seasonal pattern is 24.

3.3 MLP

The typical structure of a MLP can be illustrated as Figure 3.4. As described in Section 2.3, it has one input layer containing neurons of input variables, one hidden layer containing hidden neurons and one output layer containing neurons of input variables. Relationships between input and output variables are captured by the connections between neurons and activation functions as Equation 8,

$$y_k=\sum_{j=1}^{M}w'_{jk}y'_k+b'_k \tag{8}$$

in which,

$$y'_j=f\left(net_j\right) \tag{9}$$

$$net_j=\sum_{i=1}^{N}w_{ij}x_i+b_j \tag{10}$$

where,

y_k = value predicted by the *k*-th output neuron;

w'_{jk} = weight of the *j*-th hidden neuron to the *k*-th output neuron;

b'_k = bias added to the *k*-th output neuron;

M =a number of hidden neurons;

y'_j = value predicted by the *j*-th hidden neuron;

f = transfer function;

w_{ij} = weight of the *i*-th input neuron to the *j*-th hidden neuron;

x_i = value of the *i*-th input neuron;

b_j = bias added to the *j*-th hidden neuron;

w_{ij} = weight of the *i*-th input neuron to the *j*-th hidden neuron;

N = number of input neurons.

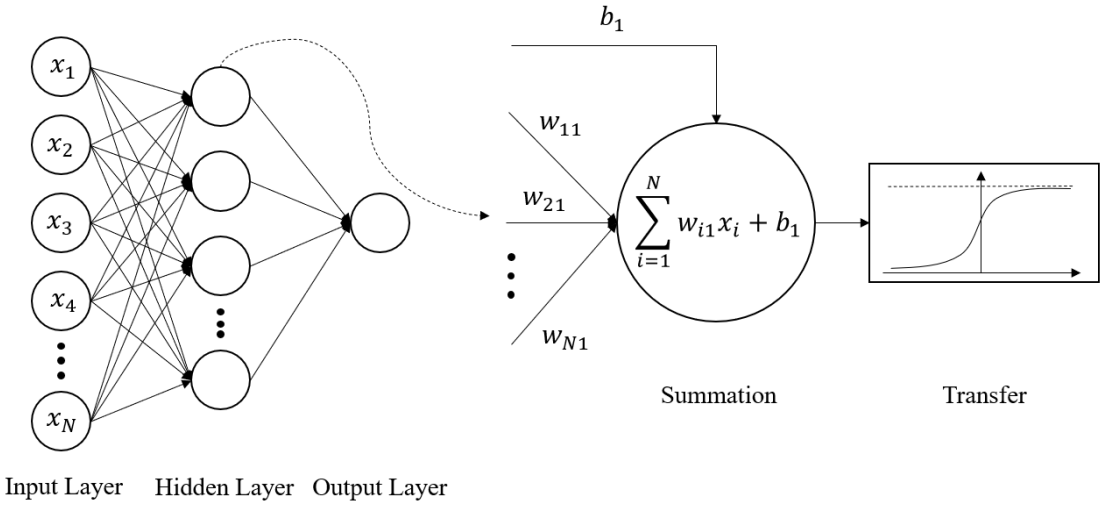


Figure 3.4: Architecture of MLP (Deng & Shi, 2022a)

Numbers of input and output neurons are decided by the numbers of input and output variables. In this project, traffic mobility of the previous 24 hours was utilized to predicted the one of the next 24 hours. Accordingly, the numbers of input and output neurons were 24. The number of hidden neurons was treated as a hyperparameter and determined via a grid search introduced in the next chapter.

MLP is widely applied as predictive model for complicated relationships between input and output variables (Deng & Shi, 2022a). Considering the number of parameters to be calibrated (i.e., weights and biases) is directly determined by the numbers of applied neurons, limited data for model training and overly complicated model structure may lead to MLP with insufficient stability and robustness (Deng & Shi, 2022a, 2022b). Therefore, in addition to model accuracy, properties such as model reproducibility, residual normality, etc. are important as well in model evaluation and were examined in this project. Details and results are included in the next chapter.

3.4 CNN and LSTM

CNN treats time series as image with time varying variable and time distributed on two axes. CNN applied in this project is one-dimensional (1D) CNN. As illustrated in Figure 3.5, it first transforms data into a 1D vector as model input. As introduced in Section 2.4, features of data are captured by the convolutional filters containing weights. They slide through the input vector to calculate the dot product in the overlapped region. The feature maps are accordingly generated as the vectors in the convolution layer in Figure 3.5. Those feature maps are further processed via activation function, pooling and flattening to the input layer (“flatten layer” in

Figure 3.5) of a fully connected NN, of which the mechanism is similar to MLP introduced in the previous section. As in MLP, length of input and output layers was 24. Filter weights and weights and biases in the fully connected NN were calibrated during the model training. In addition to the neuron number in the fully connected layer, the epoch number (i.e., the number of times that the learning algorithm works through the training dataset), batch size (i.e., the number of training examples in one forward/backward pass), etc. were treated as hyperparameters and determined via a grid search.

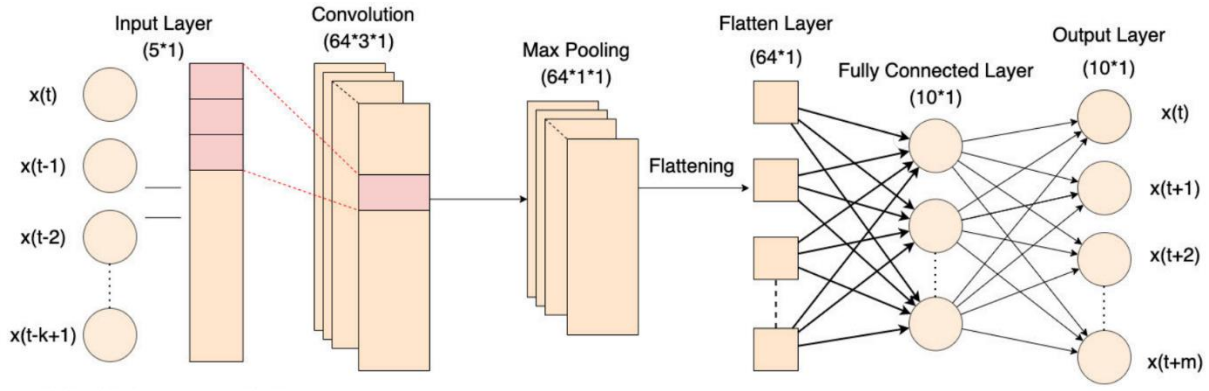


Figure 3.5: 1D CNN for time series prediction (R. Chandra, Goyal, & Gupta, 2021)

Each LSTM cell processes input of the current step X_t as well as the output cell state and output of the previous step C_{t-1} and h_{t-1}. In the current step, the activation vector of the forget gate f_t, activation vector of the input gate i_t, current input cell state $\tilde{C}_t$ and activation vector of the output gate o_t are calculated as in the following equations,

$$f_t = \sigma_g\left(W_f X_t + U_f h_{t-1} + b_f\right) \tag{11}$$

$$i_t = \sigma_g\left(W_i X_t + U_i h_{t-1} + b_i\right) \tag{12}$$

$$\tilde{C}_t = \tanh\left(W_C X_t + U_C h_{t-1} + b_C\right) \tag{13}$$

$$o_t = \sigma_g\left(W_o X_t + U_o h_{t-1} + b_o\right) \tag{14}$$

where,

σ_g = gate activation function;

W_f, W_i, W_c, W_o, U_f, U_i, U_c, U_o = weight matrices;

b_f, b_i, b_c, b_o = bias vectors;

tanh = hyperbolic tangent function.

Then, the output cell state and output of the current step C_t and h_t can be calculated as in the following equations.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{15}$$

$$h_t = o_t \cdot \tanh C_t \tag{16}$$

The output from each time step is finally interpreted by a fully connected layer (Brownlee, 2018; R. Chandra et al., 2021). Accordingly, parameters in the weight matrices and bias vectors were calibrated during the model training. The epoch number, batch size, dimension of the hidden state and neuron number in the fully connected layer were treated as hyperparameters and determined via a grid search.

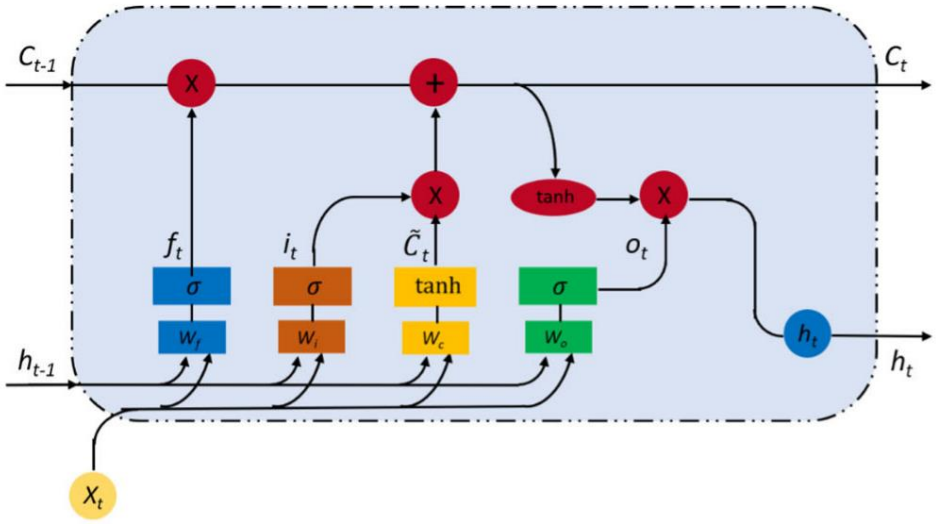


Figure 3.6: Structure of a LSTM cell (Pu, Liu, Shi, Cui, & Wang, 2020)

3.5 Summary

This chapter describes the data collection and processing and introduces the applied models for traffic mobility prediction. The main work and findings in this chapter are summarized as follows.

- The historical traffic speed and volume data was collected from a certain location of an interstate highway in the Washington state. The data of a whole month was processed to time series with the interval of one hour. For both traffic speed and volume, a total of 744 data was divided into the first 672 for model training and the rest 72 for model validation;
- ARIMA was selected as the representative of statistical models to be applied in this project. Considering the data characteristics, SARIMA with the time span of repeating seasonal pattern 24 was applied in the model construction and application. Model parameters to be determined included nonseasonal and seasonal AR, MA and differencing orders;
- MLP was selected as the representative of traditional ML models to be applied in this project. Neuron number in both input and output layers was 24. Calibrated parameters included neuron weights and biases and hyperparameters included neuron number in the hidden layer;
- CNN and LSTM were selected as the representatives of DL models to be applied in this project. Both input and output sequences had the length of 24. In CNN, calibrated parameters included filter weights, etc. and hyperparameters included epoch number, batch size, etc. In LSTM, calibrated parameters included weight matrices and bias vectors

and hyperparameters included epoch number, batch size, dimension of the hidden state, etc.

Chapter 4. Results and Analyses

This chapter presents the results and analyses of model construction and prediction. The calibrated parameters, prediction performance evaluation and comparison are provided as well. Traffic volume and speed are described separately in Section 4.1 and Section 4.2.

4.1 Traffic Volume

4.1.1 ARIMA

Before model construction, the training dataset was normalized as in the following equation for a clearer model performance evaluation and comparison.

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{17}$$

where,

x' = normalized value in the range [0, 1];
x = original value;
$x_{\max}$, $x_{\min}$ = maximum and minimum of the (training) dataset.

The software IBM SPSS Statistics (IBM, 2016) was utilized to construct SARIMA models in this project. Nonseasonal and seasonal AR, MA and differencing orders *p*, *q*, *d* and *P*, *Q*, *D* were determined in the model construction.

Basic information of the constructed model using the training dataset is provided in Table 4.1 including model parameter values and fitting performance. R^2, RMSE, MAPE, MaxAPE, MAE, MaxAE and Normalized BIC in Table 4.1(b) stand for coefficient of determination, root mean square error, mean absolute percentage error, maximum absolute percentage error, mean absolute error, maximum absolute error and normalized Bayesian Information Criterion, respectively. Their mathematical expressions are provided in the following equations. Specifically, BIC is a typical information criterion score reflecting the trade-off between goodness-of-fit and model complexity (Cohen & Berchenko, 2021; Schwarz, 1978). In addition to typical indicators mentioned above, the comparison between measured and fitted training dataset is presented in Figure 4.1 for a straightforward illustration of fitting performance.

$$R^2 = 1 - \frac{\sum_i \left(y_i - \hat{y}_i\right)^2}{\sum_i \left(y_i - \bar{y}\right)^2} \tag{18}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} \left(y_i - \hat{y}_i\right)^2}{N}} \tag{19}$$

$$MAPE = \frac{1}{N} \sum_{i=1}^{N} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \tag{20}$$

$$MaxAPE = \max\left\{\left|\frac{y_i - \hat{y}_i}{y_i}\right|\right\} \tag{21}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{22}$$

$$MaxAE = \max\left\{|y_i - \hat{y}_i|\right\} \tag{23}$$

$$BIC = -2\log \mathrm{L}\left(\hat{\theta}\right) + k\log N \tag{24}$$

where,

y_i = the i-th component of the actual value vector;

$\hat{y}_i$ = the i-th component of the predicted value vector;

$\bar{y}$ = mean of all actual values;

N = number of components in the actual (or predicted) value vector;

k = number of model parameters;

θ = a set of all model parameters;

$\mathrm{L}\left(\hat{\theta}\right)$ = the maximized likelihood of the candidate model.

Table 4.1: Information of the fitting model including (a) model parameters and (b) model evaluation

(a)

Parameter	**Value**
ARIMA(*p*,*d*,*q*)(*P*,*D*,*Q*)$_m$	*ARIMA*(*2*,*1*,*6*)(*1*,*1*,*0*)$_{24}$
AR.L1	0.355
AR.L2	-0.444
MA.L2	-0.686
MA.L6	0.173
AR.S.L1	-0.250

(b)

Indicator	**Value**
R^2	0.9662
RMSE	0.0523
MAPE	17.8100
MaxAPE	343.3066
MAE	0.0377
MaxAE	0.2440
Normalized BIC	-5.8532

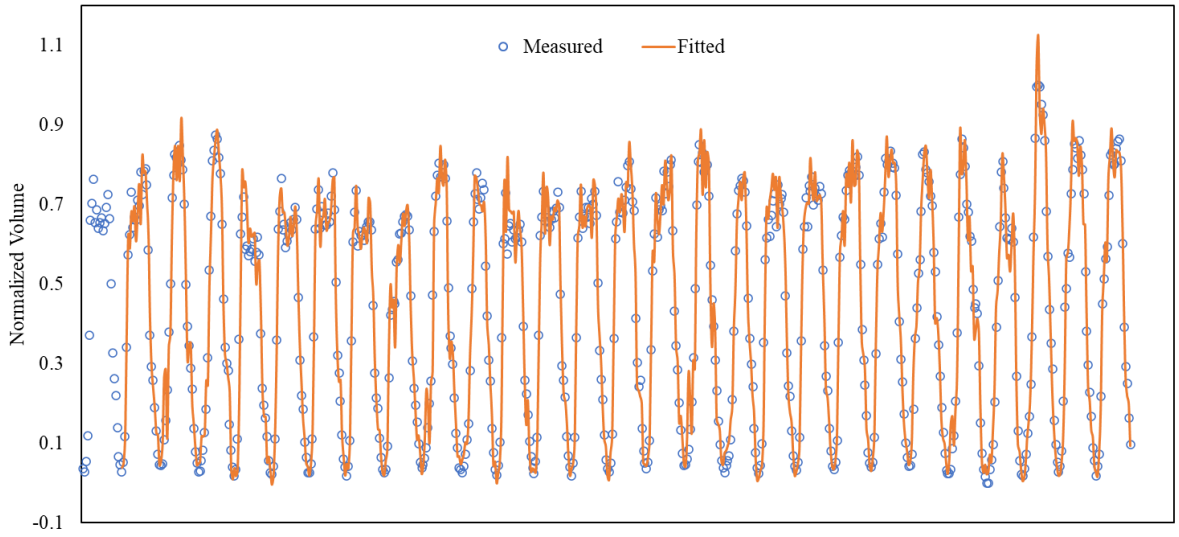


Figure 4.1: Comparison between measured and fitted data

As mentioned in Section 3.3, ACF and PACF provide correlations between two variables in time series. Considering model residuals are supposed to be random, ACF and PACF values for model residuals are better insignificant. Otherwise, AR and MA orders should be adjusted accordingly. Figure 4.2 shows the ACF and PACF values for residuals of the constructed SARIMA model.

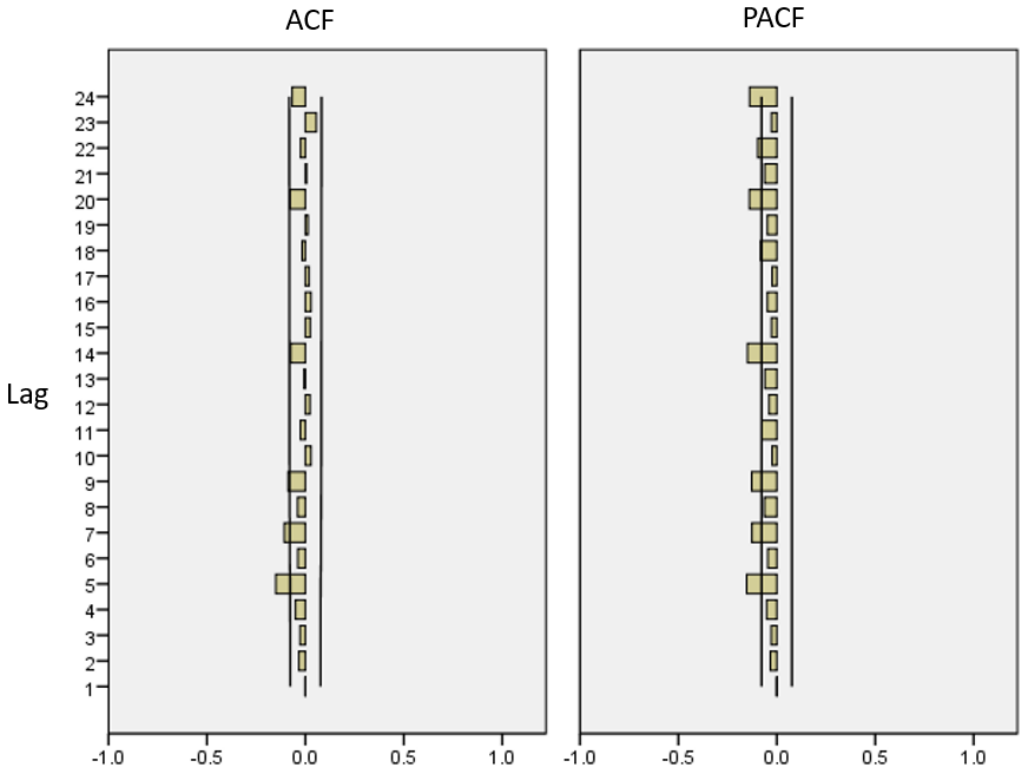


Figure 4.2: ACF and PACF plots of residuals

Similar to the training dataset, prediction performance of the constructed model can be evaluated with the validation dataset. Table 4.2 presents four typical indicators for prediction accuracy and Figure 4.3 presents the comparison of measured and predicted validation dataset. For both training and validation datasets, SARIMA model captured the development pattern of traffic volume with basic accuracy, as indicated by Table 4.1(b), Figure 4.1, Table 4.2 and Figure 4.3.

Table 4.2: Prediction performance of the constructed SARIMA model

Indicator	Value
R^2	0.8863
RMSE	0.1093
MAPE	33.6398
MAE	0.0675

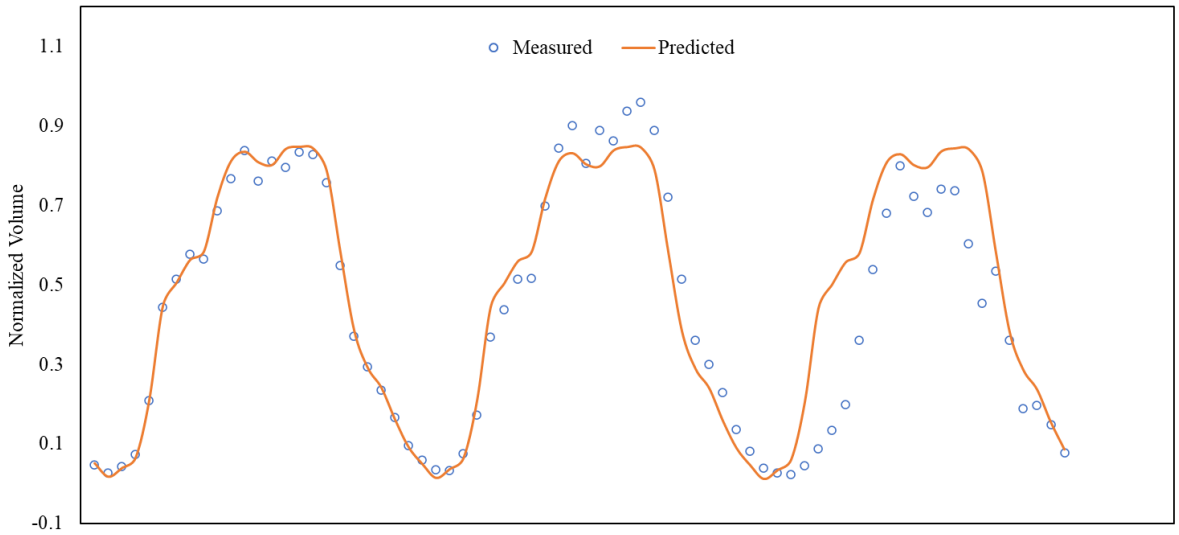


Figure 4.3: Comparison between measured and predicted data

4.1.2 MLP, CNN and LSTM

As described in Section 3.4 and Section 3.5, hyperparameters of MLP, CNN and LSTM models were determined via grid search with the training dataset and RMSE as the indicator (Brownlee, 2018). Information of constructed models including hyperparameter values is presented in

Table 4.3. Specifically, the optimizer “Adam” refers to the adaptive moment estimation (Adam) algorithm (Kingma & Ba, 2014) and activation function “ReLU” refers to the Rectified Linear Unit (Fukushima, 1969).

Table 4.3: Information of constructed MLP, CNN and LSTM models

MLP		CNN		LSTM	
Parameter	**Value**	**Parameter**	**Value**	**Parameter**	**Value**
Hidden Neuron Number	107	Neuron Number in the Fully Connected Layer	30	Hidden State Dimension	170
		Filter Number	16	Neuron Number in the Fully Connected Layer	30
		Filter Size	3		
		Pool Size	2		
Batch Size	100	Batch Size	130	Batch Size	20
Epoch Number	100	Epoch Number	100	Epoch Number	80
Optimizer	Adam	Optimizer	Adam	Optimizer	Adam
Activation Function	ReLU	Activation Function	ReLU	Activation Function	ReLU

Different from ARIMA model, model forms of ML and DL models are difficult to express explicitly. Model performance in terms of model accuracy, stability, etc. can be evaluated via statistical methods and comparisons with traditional models (Deng & Shi, 2022a, 2022b). As for model stability, it was considered in traditional ML and DL models for the possible case in which models of significant differences are generated via repetitive model constructions. It results from overly complicated model structure (excessive calibrated parameters) compared with limited information provided by the training data. Accordingly, 10 repetitive model constructions were conducted for each MLP, CNN and LSTM model in this project; and indicators of model performance were expressed with confidence intervals. The confidence level selected in this project was 95% and the confidence interval can be calculated as follows.

$$CI = \bar{x} \pm z \frac{s}{\sqrt{n}} \tag{25}$$

where,

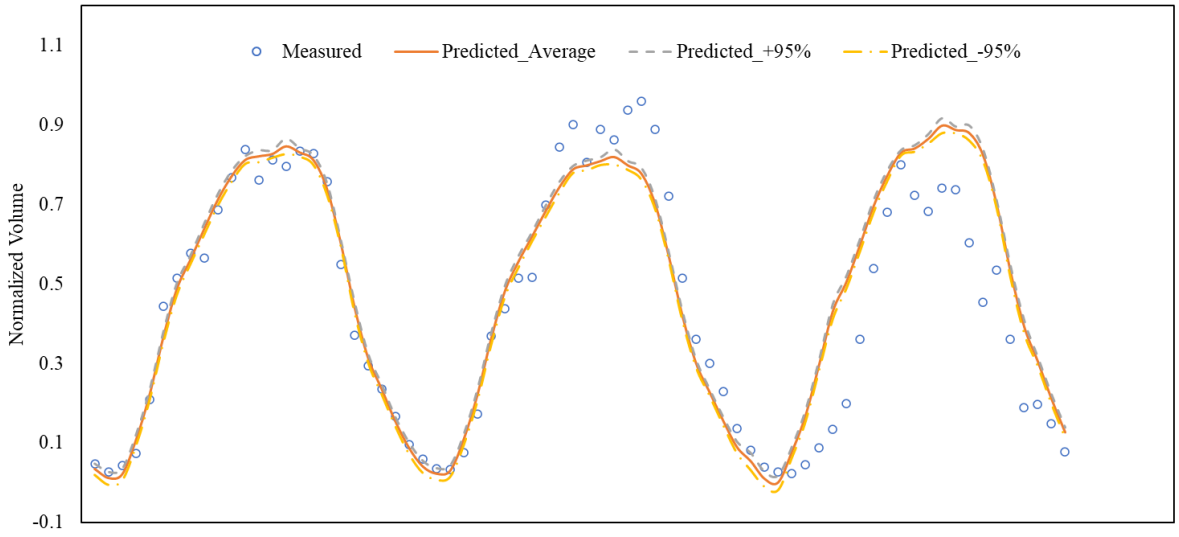
CI = confidence interval;
$\bar{x}$ = sample mean;
z = confidence level value taking 1.960 for 95% confidence level;
s = sample standard deviation;
n = sample size taking 10 in this project.

Table 4.4 presents the prediction performance of the constructed MLP, CNN and LSTM models with the validation dataset. R^2, RMSE, MAPE, and MAE were selected as the indicators reflecting the averaged accuracy within the validation dataset. Indicator values reflecting higher accuracy (higher R^2 and lower RMSE, MAPE, and MAE) and stability (narrower confidence

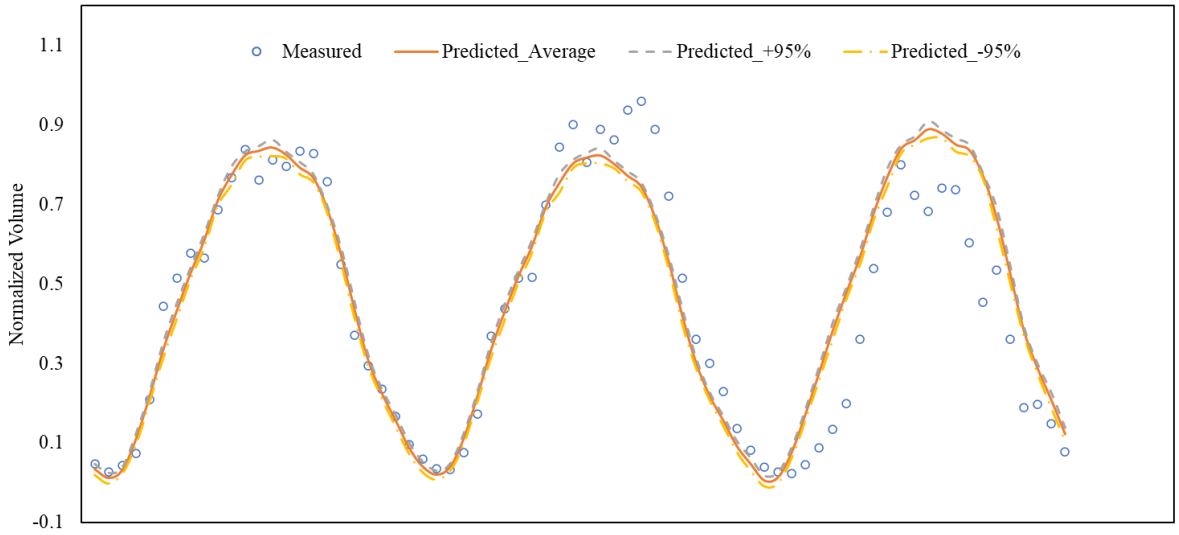
interval) are marked in bold in Table 4.4. It can be seen that differences in prediction accuracy and stability between three models are not obvious. However, LSTM model achieved relatively higher accuracy and stability in most cases. Compared with the prediction performance of SARIMA model indicated in Table 4.2, there is no advantage of applying traditional ML and DL models in predicting traffic mobility in this project. Figure 4.4 presents the comparison of measured and predicted validation dataset with confidence intervals for a clearer illustration.

Table 4.4: Prediction performance of the constructed MLP, CNN and LSTM models

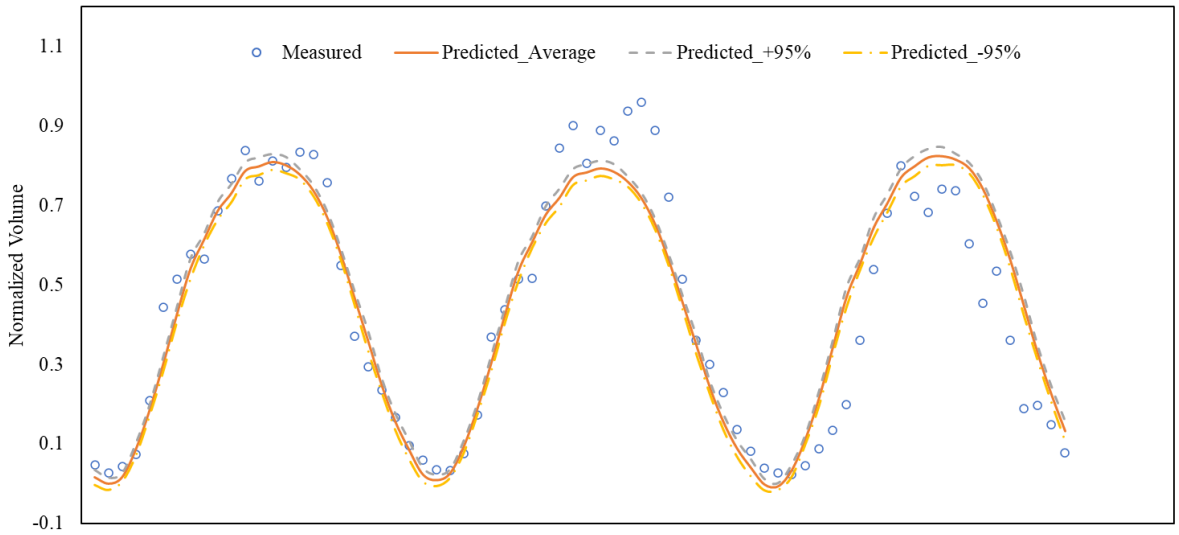
Indicator	**Value**		
	MLP	CNN	LSTM
R^2	0.8655± 0.0033	0.8710±0.0044	**0.8730±0.0026**
RMSE	0.1183±0.0018	0.1124±0.0026	**0.1104±0.0013**
MAPE	40.2066±2.3275	37.8462±**2.2920**	**36.7338**±2.9234
MAE	0.0850±**0.0016**	0.0833±0.0025	**0.0832**±0.0021



(a)



(b)



(c)

Figure 4.4: Comparison between measured and predicted data using (a) MLP, (b) CNN and (c) LSTM

Finally, normality of model residuals were checked with the validation dataset using Shapiro-Wilk test (Shapiro & Wilk, 1965). It tests the null hypothesis that there is no difference between the tested sample and a normal distribution using the statistic *W* expressed in the following equation.

$$W = \frac{\left(\sum_{i=1}^{n} a_i x_{(i)}\right)^2}{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2} \tag{26}$$

where,

$x_{(i)}$ = the *i*-th ordered sample values;

a_i = the i-th constant generated from the mean, variance and covariance of n order statistics from a normal distribution;
$\bar{x}$ = sample mean;
n = sample size taking 72 in this project.

Table 4.5 presents the test results of all constructed MLP, CNN and LSTM models from 10 repetitive mode constructions. Compared with MLP models, residuals of most CNN and LSTM models with the validation dataset achieved a normal distribution at the 95% confidence level.

Table 4.5: Shapiro-Wilk test results of constructed MLP, CNN and LSTM models

Case No.	W Value		
	MLP	CNN	LSTM
1	0.9337	0.9711	0.9718
2	0.9624	0.9731	0.9593
3	0.9548	0.9813	0.9804
4	0.9529	0.9700	0.9676
5	0.9602	0.9792	0.9649
6	0.9431	0.9739	0.9690
7	0.9551	0.9721	0.9782
8	0.9468	0.9671	0.9799
9	0.9705	0.9649	0.9716
10	0.9624	0.9608	0.9726
Passing Ratio	10%	80%	80%
Test Information	Accepted range of the statistic W at a 95% confidence level: [0.9662, 1.0000]		

4.2 Traffic Speed

4.2.1 ARIMA

As in Section 4.2, basic information of the constructed SARIMA model using the training dataset of traffic speed is provided in Table 4.6; and the comparison between measured and fitted training dataset is presented in Figure 4.5. Compared with traffic mobility, majority of traffic speed is more randomly distributed in a narrower range and daily pattern is not obvious in the data. Indicator values in Table 4.6(b) are of a similar level as those of the SARIMA model for traffic volume. ACF and PACF plots of residuals indicate there is no significant correlation between variables in the constructed model.

Table 4.6: Information of the fitting model including (a) model parameters and (b) model evaluation

(a)

Parameter	Value
$ARIMA(p,d,q)(P,D,Q)_m$	$ARIMA(3,0,1)(1,0,1)_{24}$
AR.L1	-0.236
AR.L2	0.807
AR.L3	0.082
MA.L1	-0.988
AR.S.L1	0.982
MA.S.L1	0.899
Constant	0.754

(b)

Indicator	Value
R^2	0.699
RMSE	0.062
MAPE	6.109
MaxAPE	138.138
MAE	0.043
MaxAE	0.447
Normalized BIC	-5.494

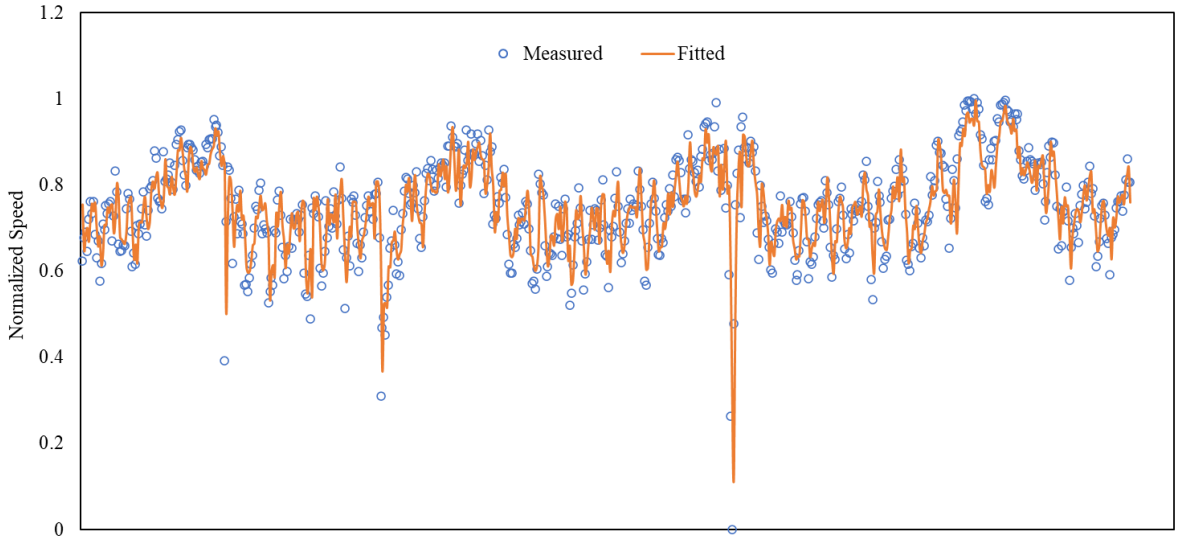


Figure 4.5: Comparison between measured and fitted data

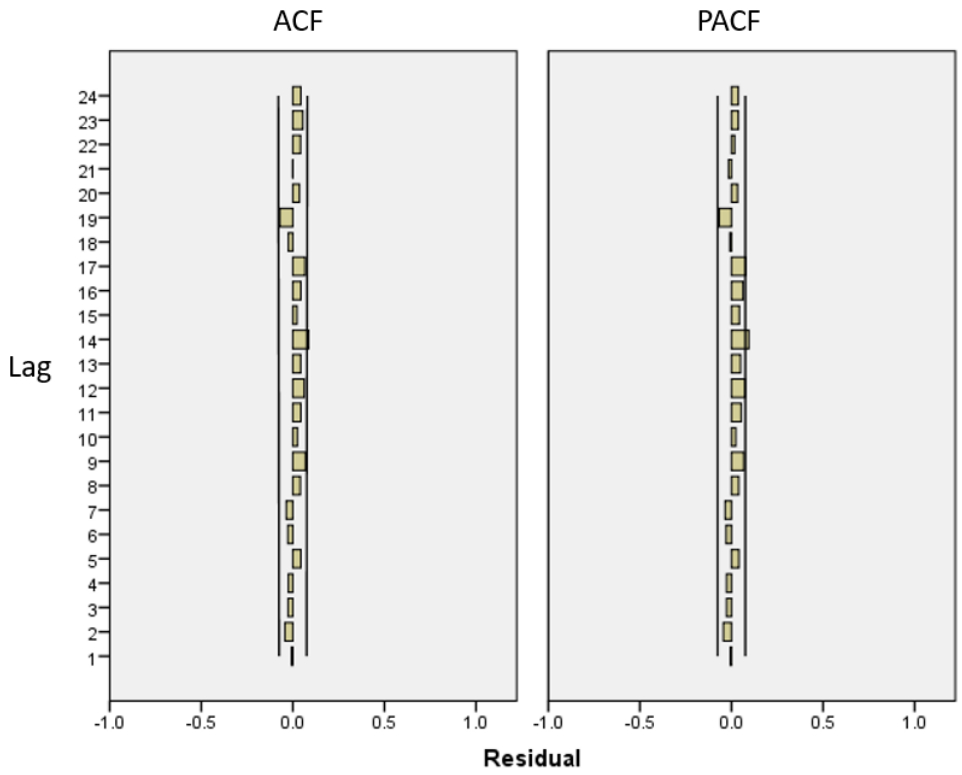


Figure 4.6: ACF and PACF plots of residuals

Table 4.7 presents values of indicators for prediction accuracy and Figure 4.7 presents the comparison of measured and predicted validation dataset. Different from the fitting performance, much poorer prediction accuracy was achieved by the constructed SARIMA model and is reflected on the low value of R^2. Due to the narrow range of traffic speed data distribution, RMSE, MAPE and MAE were limitedly affected.

Table 4.7: Prediction performance of the constructed SARIMA model

Indicator	Value
R^2	0.0461
RMSE	0.1134
MAPE	12.2811
MAE	0.0804

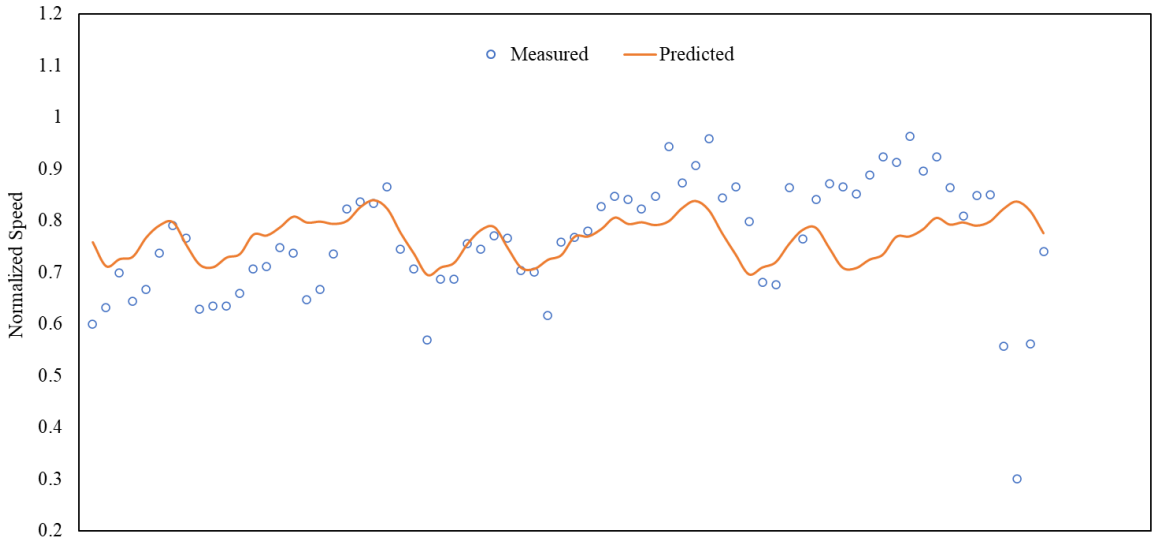


Figure 4.7: Comparison between measured and predicted data

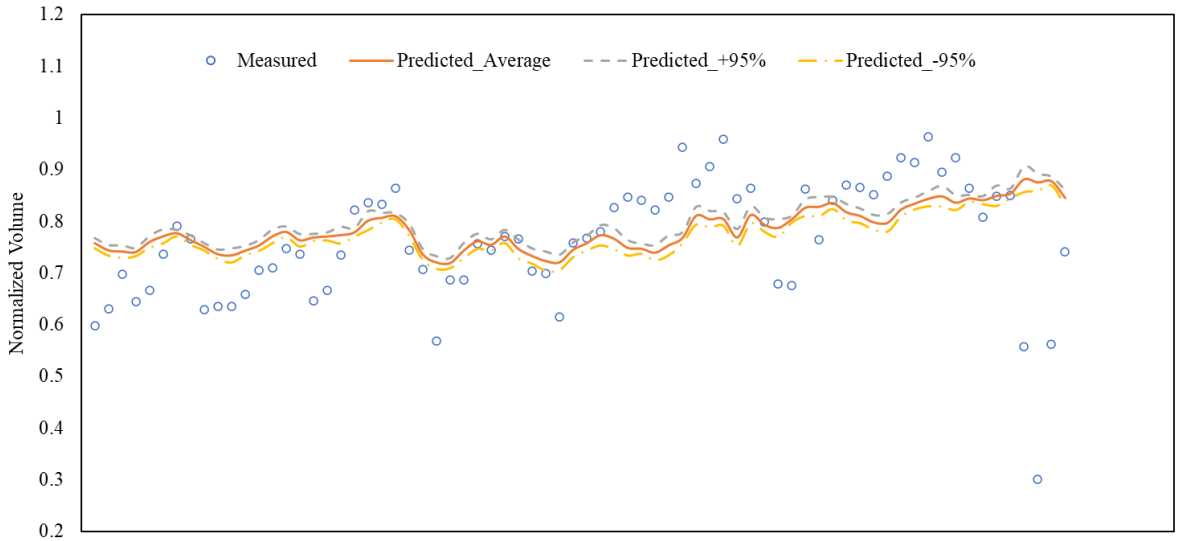
4.2.2 MLP, CNN and LSTM

For simplicity, hyperparameter values of MLP, CNN and LSTM models for traffic speed were directly taken as those in

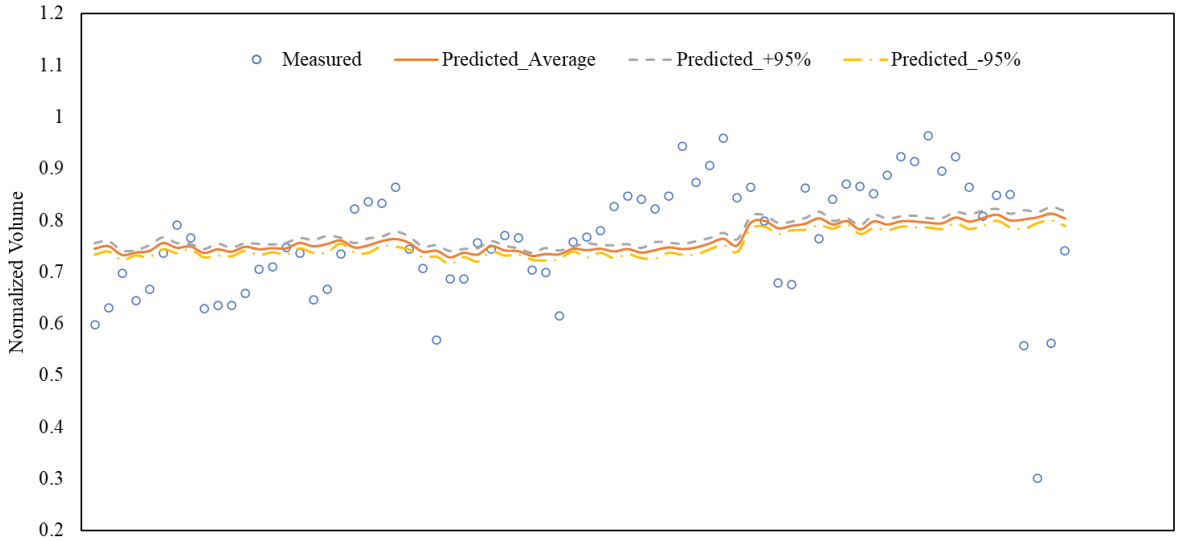
Table 4.3. Table 4.8 presents the prediction performance of the constructed MLP, CNN and LSTM models with the validation dataset; and indicator values reflecting higher accuracy and stability are marked in bold. Similar to SARIMA model, prediction performance of all constructed MLP, CNN and LSTM models is not desirable. Besides, no significant difference exists in prediction accuracy and stability between three model types. In general, neither statistical nor ML models can capture the development pattern with model settings in this project. Possible reasons can be that first, hyperparameter values of ML models for traffic speed are very different from those for traffic volume. New gris search should be conducted; Second, time span of the repeating pattern is not 24 (i.e., daily) for traffic speed as traffic volume. Even there exist no certain pattern in traffic speed data; Third, traffic speed is affected by other important factors such as road surface and network conditions, driver's behaviors, weather condition, etc. Different model types should be applied for capturing and predicting traffic speed development.

Table 4.8: Prediction performance of the constructed MLP, CNN and LSTM models

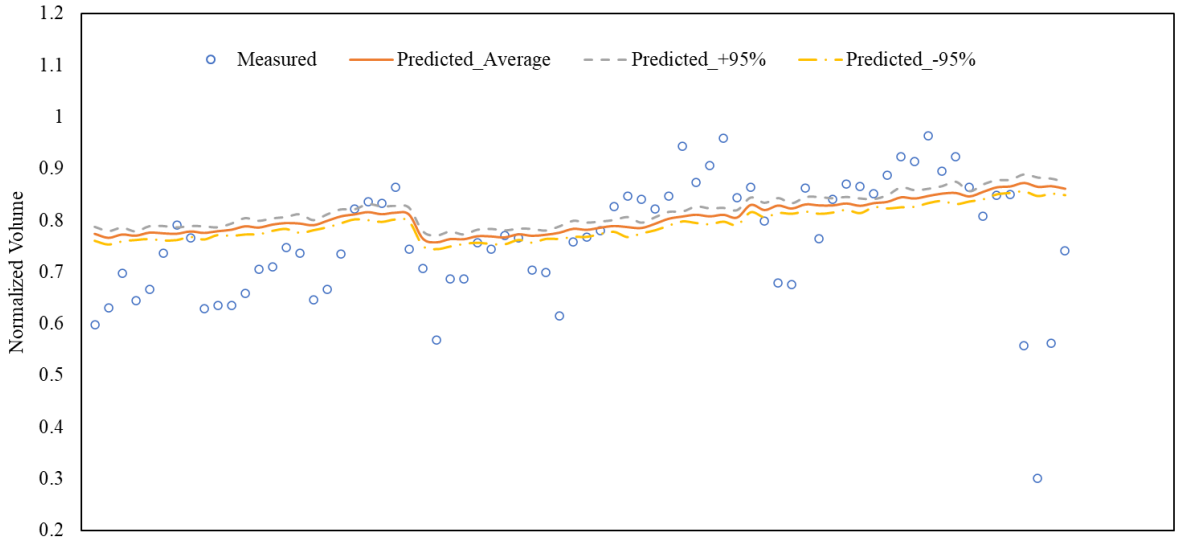
Indicator	**Value**		
	MLP	CNN	LSTM
R^2	0.0521±**0.0099**	0.0435±0.0162	**0.0689**±0.0108
RMSE	0.1162±0.0018	**0.1134**±**0.0013**	0.1201±0.0032
MAPE	**12.5791**±0.1920	12.8586±**0.1416**	13.4442±0.3780
MAE	**0.0793**±0.0011	0.0853±**0.0008**	0.0838±0.0020


Measured
Predicted_Average
Predicted_+95%
Predicted_-95%
Normalized Volume
1.2
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2

(a)


Measured
Predicted_Average
Predicted_+95%
Predicted_-95%
Normalized Volume
1.2
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2

(b)



(c)

Figure 4.8: Comparison between measured and predicted data using (a) MLP, (b) CNN and (c) LSTM

Table 4.9 presents the test results of all constructed MLP, CNN and LSTM models from 10 repetitive mode constructions. Results show that the difference between the residual distribution of all models and normal distribution is great enough to be statistically significant at a 95% confidence level.

Table 4.9: Shapiro-Wilk test results of constructed MLP, CNN and LSTM models

Case No.	**W Value**		
	MLP	CNN	LSTM
1	0.8752	0.9110	0.9055
2	0.8565	0.9298	0.9327
3	0.8823	0.9242	0.8559
4	0.8578	0.9144	0.8567
5	0.8668	0.9131	0.8682
6	0.8206	0.9039	0.8595
7	0.8464	0.9258	0.8811
8	0.8911	0.8986	0.9130
9	0.8424	0.9187	0.9060
10	0.8652	0.9258	0.8853
Passing Ratio	0%	0%	0%
Test Information	Accepted range of the statistic W at a 95% confidence level: [0.9662, 1.0000]		

4.3 Summary

This chapter presents results and analyses of model construction and prediction. The information of data normalization, hyperparameter determination, model fitting and prediction performance

evaluation and comparison is provided. The main findings in this chapter can be summarized as follows.

- The SARIMA model achieved satisfactory performance in both fitting and predicting traffic volume. As for fitting accuracy with the training dataset, the R^2, RMSE, MAPE and MAE are 0.9662, 0.0523, 17.8100 and 0.0377, respectively. The ACF and PACF values for model residuals indicate no significant correlations between variables in the constructed SARIMA model. As for prediction accuracy with the validation dataset, the R^2, RMSE, MAPE and MAE are 0.8863, 0.1093, 33.6398 and 0.0675, respectively.
- The MLP, CNN and LSTM models achieved similar level of fitting and predicting performance as SARIMA model for traffic volume. In addition, these ML models gained desirable stability with performance indicators maintained a similar level in 10 repetitive model constructions. For example, the R^2, RMSE, MAPE and MAE with 95% confidence interval of LSTM model are 0.8730±0.0026, 0.1104±0.0013, 36.7338±2.9234 and 0.0832±0.0021, respectively. Moreover, the Shapiro-Wilk test showed that the residuals of most constructed MLP, CNN and LSTM models with the validation dataset followed a normal distribution at the 95% confidence level.
- Traffic speed is distributed more randomly in the range [55, 70] (mph) and no obvious pattern can be observed from the data. Therefore, compared with traffic volume, both fitting and prediction performance of SARIMA model was poorer for traffic speed. As for fitting accuracy with the training dataset, the R^2, RMSE, MAPE and MAE are 0.699, 0.062, 6.109 and 0.043, respectively. As for prediction accuracy, values of these indicators are 0.0461, 0.1134, 12.2811 and 0.0804, respectively.
- The prediction performance was not improved by applying MLP, CNN and LSTM models. For example, the R^2, RMSE, MAPE and MAE with 95% confidence interval of LSTM model are 0.0689±0.0108, 0.1201±0.0032, 13.4442±0.3780 and 0.0838±0.0020, respectively.

Chapter 5. Conclusions and Recommendations

To develop predictive models for traffic mobility using ML approaches, this project first reviewed current predictive models for predicting traffic mobility components - traffic speed and traffic volume. Then, according to the characteristics of collected data, appropriate statistical, traditional ML and DL models were selected and introduced with focuses on the mathematical form, structure, and construction procedures. Next, the collected data was processed and divided into the training and validation datasets to construct, validate and compare model fitting and prediction performance. Finally, the results of model evaluation and comparison were analyzed and summarized for their advantages and limitations in future applications. The flow of major work in this project can be summarized as in Figure 5.1.

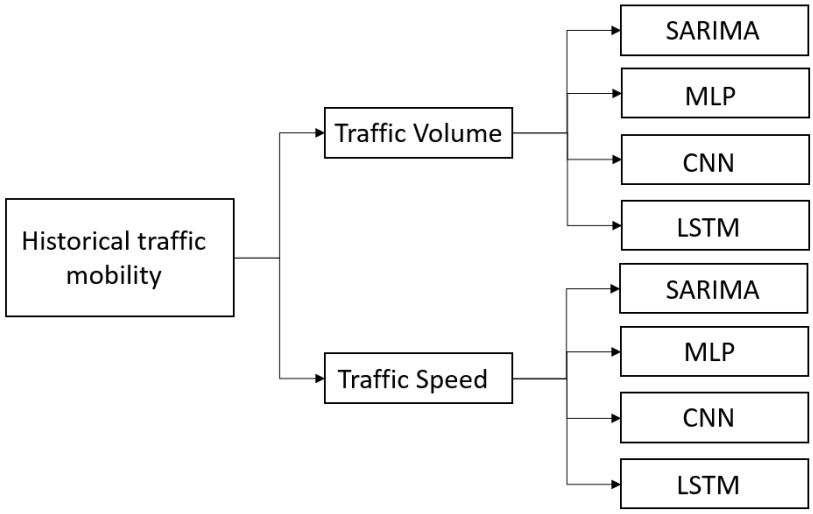


Figure 5.1: Flow of work in this project

5.1 Conclusions

This section presents a summary of conclusions from this study, as follows.

Chapter 2 briefly reviews the predictive models for traffic mobility in terms of traffic speed and volume/flow. The focused model types are statistical, traditional ML and DL models. Different model types have different model structures and algorithms. Accordingly, they can capture different characteristics (spatial, temporal or spatio-temporal dependencies) of traffic mobility. They should be selected with the consideration of the characteristics of applied data.

Chapter 3 describes the data collection and processing and introduces applied predictive models for traffic mobility prediction. Collected data was processed to time series with the interval of one hour for both traffic speed and volume. Accordingly, ARIMA, MLP, CNN and LSTM were selected as the representatives of statistical, traditional ML and DL models to be applied for data fitting and prediction. Each model has its own hyperparameters, input and output types and calibrated parameters to be determined in the model preparation and training.

Chapter 4 presents the results and analyses of model construction and prediction. Information of data normalization, hyperparameter determination, model fitting and prediction performance evaluation and comparison is provided. In general, SARIMA, MLP, CNN and LSTM achieved similar and desirable performance in both fitting and predicting traffic volume development. The performance was represented by typical indicators such as R^2, RMSE, MAPE and MAE for model accuracy, ACF and PACF values for model residual correlations, confidence interval for model stability and Shapiro-Wilk statistic for residual normality. In contrast, the performance of these models was poorer for traffic speed, possibly due to its data characteristics such as distribution and dependencies.

5.2 Recommendations

The recommendations for future research are summarized as follows.

- Considering that neither statistical nor ML models applied in this project can desirably capture the development pattern of traffic speed, possible strategies should be conducted such as finding new hyperparameter values of ML models, reducing the time span of the repeating pattern, adopting new model types and/or influential factors such as road surface and network conditions, driver's behaviors, and weather conditions in the model construction.
- Modelling and prediction techniques provided in this project can serve as tools in investigating and quantifying macroscopic and microscopic effects of various factors on the traffic mobility.

References

1. Akaike, H. (1974). A new look at the statistical model identification. IEEE Transactions on Automatic Control, 19(6), 716-723.
2. Anciaes, P. R., Metcalfe, P. J., & Heywood, C. (2017). Social impacts of road traffic: Perceptions and priorities of local residents. Impact Assessment and Project Appraisal, 35(2), 172-183.
3. Bao, X., Jiang, D., Yang, X., & Wang, H. (2021). An improved deep belief network for traffic prediction considering weather factors. Alexandria Engineering Journal, 60(1), 413-420.
4. Box, G. E., & Jenkins, G. M. (1970). Time Series Analysis: Forecasting and Control. San Francisco, CA: Holden-Day, Inc.
5. Brownlee, J. (2018). Deep learning for time series forecasting: predict the future with MLPs, CNNs and LSTMs in Python. In.
6. Cao, M., Li, V. O., & Chan, V. W. (2020). A CNN-LSTM model for traffic speed prediction. Paper presented at the 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), Online.
7. Chandra, R., Goyal, S., & Gupta, R. (2021). Evaluation of deep learning models for multi-step ahead time series prediction. IEEE Access, 9, 83105-83123.
8. Chandra, S. R., & Al-Deek, H. (2009). Predictions of freeway traffic speeds and volumes using vector autoregressive models. Journal of Intelligent Transportation Systems, 13(2), 53-72.
9. Chen, C., & Shi, X. (2019). Modeling the Macroscopic Effects of Winter Maintenance Operations on Traffic Mobility on Washington Highways. Retrieved from Charlotte, NC: https://cammse.uncc.edu/sites/cammse.uncc.edu/files/media/CAMMSE-UNCC-2018-UTC-Project-Report-17-Shi-Final.pdf
10. Cho, K., Van Merriënboer, B., Bahdanau, D., & Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. arXiv:1409.1259.
11. Cohen, N., & Berchenko, Y. (2021). Normalized information criteria and model selection in the presence of missing data. Mathematics, 9(19), 2474.
12. Dailey, D. J. (1999). A statistical algorithm for estimating speed from single loop volume and occupancy measurements. Transportation Research Part B: Methodological, 33(5), 313-322.
13. Deng, Y., & Shi, X. (2022a). An accurate, reproducible and robust model to predict the rutting of asphalt pavement: neural networks coupled with particle swarm optimization. IEEE Intelligent Transportation Systems Transactions. doi:10.1109/TITS.2022.3149268
14. Deng, Y., & Shi, X. (2022b). Development of predictive models of asphalt pavement distresses in Idaho through gene expression programming. Neural Computing and Applications(34), 14913–14927.
15. Dickey, D. A., & Fuller, W. A. (1979). Distribution of the estimators for autoregressive time series with a unit root. Journal of the American Statistical Association, 74(366a), 427-431.

16. Do, L. N., Taherifar, N., & Vu, H. L. (2019). Survey of neural network-based models for short-term traffic state prediction. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 9(1), e1285.
17. Fu, R., Zhang, Z., & Li, L. (2016). Using LSTM and GRU neural network methods for traffic flow prediction. Paper presented at the 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), Wuhan, China.
18. Fukushima, K. (1969). Visual feature extraction by a multilayered network of analog threshold elements. IEEE Transactions on Systems Science and Cybernetics, 5(4), 322-333.
19. Gorinova, M. I., Gordon, A. D., Sutton, C., & Vákár, M. (2021). Conditional independence by typing. ACM Transactions on Programming Languages and Systems, 44(1), 1-54.
20. Goudarzi, F. (2018). Travel time prediction: comparison of machine learning algorithms in a case study. Paper presented at the 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS).
21. Guo, J., Xia, J., & Smith, B. L. (2009). Kalman filter approach to speed estimation using single loop detector measurements under congested conditions. Journal of Transportation Engineering, 135(12), 927-934.
22. Hinton, G. E., Osindero, S., & Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. Neural Computation, 18(7), 1527-1554.
23. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735-1780.
24. Huang, W., Song, G., Hong, H., & Xie, K. (2014). Deep architecture for traffic flow prediction: deep belief networks with multitask learning. IEEE Transactions on Intelligent Transportation Systems, 15(5), 2191-2201.
25. Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: Principles and Practice (2nd ed.). Melbourne, Australia: OTexts.
26. IBM. (2016). IBM SPSS Statistics for Windows (Version 24.0.). Armonk, NY: IBM Corp.
27. Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. Electronic Markets, 31(3), 685-695.
28. Julier, S. J., & Uhlmann, J. K. (1997). New extension of the Kalman filter to nonlinear systems. Paper presented at the Signal processing, sensor fusion, and target recognition VI, Orlando, FL.
29. Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. Journal of Basic Engineering, 82(1), 35-45.
30. Kim, Y., Wang, P., Zhu, Y., & Mihaylova, L. (2018). A capsule network for traffic speed prediction in complex road networks. Paper presented at the 2018 Sensor Data Fusion: Trends, Solutions, Applications (SDF), Bonn, Germany.
31. Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint. doi:10.48550/ARXIV.1412.6980

32. Kohonen, T. (1990). The self-organizing map. Proceedings of the IEEE, 78(9), 1464-1480.
33. Kumar, K., Parida, M., & Katiyar, V. (2013). Short term traffic flow prediction for a non urban highway using artificial neural network. Procedia - Social and Behavioral Sciences, 104, 755-764.
34. Kumar, K., Parida, M., & Katiyar, V. K. (2015). Short term traffic flow prediction in heterogeneous condition using artificial neural network. Transport, 30(4), 397-405.
35. Kwiatkowski, D., Phillips, P. C., Schmidt, P., & Shin, Y. (1992). Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root? Journal of Econometrics, 54(1-3), 159-178.
36. LeCun, Y., & Bengio, Y. (1995). Convolutional networks for images, speech, and time series. In The Handbook of Brain Theory and Neural Networks. Cambridge, MA: MIT Press.
37. Li, L., Qin, L., Qu, X., Zhang, J., Wang, Y., & Ran, B. (2019). Day-ahead traffic flow forecasting based on a deep belief network optimized by the multi-objective particle swarm algorithm. Knowledge-Based Systems, 172, 1-14.
38. Litman, T. (2003). Measuring transportation: traffic, mobility and accessibility. ITE Journal, 73(10), 28-32.
39. Liu, Q., Li, J., & Lu, Z. (2021). ST-Tran: Spatial-temporal transformer for cellular traffic prediction. IEEE Communications Letters, 25(10), 3325-3329.
40. Ma, X., Dai, Z., He, Z., Ma, J., Wang, Y., & Wang, Y. (2017). Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction. Sensors, 17(4), 818.
41. Min, W., & Wynter, L. (2011). Real-time road traffic prediction with spatio-temporal correlations. Transportation Research Part C: Emerging Technologies, 19(4), 606-616.
42. Niu, Z., Zhong, G., & Yu, H. (2021). A review on the attention mechanism of deep learning. Neurocomputing, 452, 48-62.
43. Pan, Z., Liang, Y., Wang, W., Yu, Y., Zheng, Y., & Zhang, J. (2019). Urban traffic prediction from spatio-temporal data using deep meta learning. Paper presented at the Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, Anchorage, AK.
44. Peer, S., Koopmans, C. C., & Verhoef, E. T. (2012). Prediction of travel time variability for cost-benefit analysis. Transportation Research Part A: Policy and Practice, 46(1), 79-90.
45. Pu, Z., Liu, C., Shi, X., Cui, Z., & Wang, Y. (2020). Road surface friction prediction using long short-term memory neural network based on historical data. Journal of Intelligent Transportation Systems, 1-12.
46. Qi, Y., & Ishak, S. (2014). A Hidden Markov Model for short term prediction of traffic conditions on freeways. Transportation Research Part C: Emerging Technologies, 43, 95-111.
47. Rapant, L., Slaninová, K., Martinovič, J., & Martinovič, T. (2016). Traffic speed prediction using hidden Markov models for Czech Republic highways. In Agent and Multi-Agent Systems: Technology and Applications (Vol. 58, pp. 187-196). Switzerland: Springer, Cham.

48. Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). Learning internal representations by error propagation (ICS 8506). Retrieved from La Jolla, CA: https://apps.dtic.mil/dtic/tr/fulltext/u2/a164453.pdf
49. Russell, S. J., & Norvig, P. (2021). Artificial Intelligence: A Modern Approach (4th ed.). Hoboken, NJ: Pearson Education, Inc.
50. Schwarz, G. (1978). Estimating the dimension of a model. The Annals of Statistics, 6(2), 461-464.
51. Shapiro, S. S., & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). Biometrika, 52(3/4), 591-611.
52. Sharma, B., Kumar, S., Tiwari, P., Yadav, P., & Nezhurina, M. I. (2018). ANN based short-term traffic flow forecasting in undivided two lane highway. Journal of Big Data, 5(1), 1-16.
53. Sun, S., Zhang, C., & Yu, G. (2006). A Bayesian network approach to traffic flow forecasting. IEEE Transactions on Intelligent Transportation Systems, 7(1), 124-132.
54. Van Der Voort, M., Dougherty, M., & Watson, S. (1996). Combining Kohonen maps with ARIMA time series models to forecast traffic flow. Transportation Research Part C: Emerging Technologies, 4(5), 307-318.
55. Wang, X., Chen, C., Min, Y., He, J., Yang, B., & Zhang, Y. (2018). Efficient metropolitan traffic prediction based on graph recurrent neural network. arXiv:1811.00740.
56. Wang, Y., & Papageorgiou, M. (2005). Real-time freeway traffic state estimation based on extended Kalman filter: a general approach. Transportation Research Part B: Methodological, 39(2), 141-167.
57. Wang, Y., Papageorgiou, M., & Messmer, A. (2008). Real-time freeway traffic state estimation based on extended Kalman filter: Adaptive capabilities and real data testing. Transportation Research Part A: Policy and Practice, 42(10), 1340-1358.
58. Wang, Z., Su, X., & Ding, Z. (2020). Long-term traffic prediction based on lstm encoder-decoder architecture. IEEE Transactions on Intelligent Transportation Systems, 22(10), 6561-6571.
59. Williams, B. M., & Hoel, L. A. (2003). Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results. Journal of Transportation Engineering, 129(6), 664-672.
60. Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Philip, S. Y. (2020). A comprehensive survey on graph neural networks. IEEE Transactions on Neural Networks and Learning Systems, 32(1), 4-24.
61. Xiao, H., Sun, H., & Ran, B. (2004). Special factor adjustment model using fuzzy-neural network in traffic prediction. Transportation Research Record, 1879(1), 17-23.
62. Ye, Z., Zhang, Y., & Middleton, D. R. (2006). Unscented Kalman filter method for speed estimation using single loop detector data. Transportation Research Record, 1968(1), 117-125.
63. Yin, X., Wu, G., Wei, J., Shen, Y., Qi, H., & Yin, B. (2021). Deep learning on traffic prediction: Methods, analysis and future directions. IEEE Transactions on Intelligent Transportation Systems. doi:10.1109/TITS.2021.3054840

64. 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. Transportmetrica A: transport science, 15(2), 1688-1711.
65. Zhang, Y., & Huang, G. (2018). Traffic flow prediction model based on deep belief network and genetic algorithm. IET Intelligent Transport Systems, 12(6), 533-541.
66. Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., . . . Li, H. (2019). T-GCN: A temporal graph convolutional network for traffic prediction. IEEE Transactions on Intelligent Transportation Systems, 21(9), 3848-3858.
67. Zhao, Z., Chen, W., Wu, X., Chen, P. C., & Liu, J. (2017). LSTM network: a deep learning approach for short-term traffic forecast. IET Intelligent Transport Systems, 11(2), 68-75.
68. Zhou, Z., Yang, Z., Zhang, Y., Huang, Y., Chen, H., & Yu, Z. (2022). A comprehensive study of speed prediction in transportation system: From vehicle to traffic. Iscience, 25(3), 103909.
69. Zhu, Z., Peng, B., Xiong, C., & Zhang, L. (2016). Short-term traffic flow prediction with linear conditional Gaussian Bayesian network. Journal of Advanced Transportation, 50(6), 1111-1123.