RESEARCH

UDOT

Keeping Utah Moving

Utah Department of Transportation - Research Division
4501 South 2700 West - P.O. Box 148410 - SLC,UT 84114-8410

Report No. UT-21.03

UTILIZING MACHINE LEARNING TO CROSS-CHECK TRAFFIC DATA AND UNDERSTAND URBAN MOBILITY

Prepared For:

Utah Department of Transportation
Research & Innovation Division

Final Report
March 2021

DISCLAIMER

The authors alone are responsible for the preparation and accuracy of the information, data, analysis, discussions, recommendations, and conclusions presented herein. The contents do not necessarily reflect the views, opinions, endorsements, or policies of the Utah Department of Transportation or the U.S. Department of Transportation. The Utah Department of Transportation makes no representation or warranty of any kind, and assumes no liability therefore.

ACKNOWLEDGMENTS

The authors acknowledge the Utah Department of Transportation (UDOT) for funding this research, and the following individuals from UDOT and the Utah Department of Technology Services (DTS) on the Technical Advisory Committee for helping to guide the research:

- Rikki Sonnen – UDOT Engineering Manager (Traffic Management emphasis)
- Paul Jencks – DTS Information Technology Services Analyst
- Rudy Zamora – DTS Information Technology Manager
- Nicolas Black – UDOT Business Analyst Supervisor
- Kaitlin Marousis – UDOT Transportation Project Manager (Geographic Information Systems emphasis)
- Travis Jensen – UDOT Project Manager (Consultant)

TECHNICAL REPORT ABSTRACT

1. Report No. UT-21.03	2. Government Accession No. N/A	3. Recipient's Catalog No. N/A
4. Title and Subtitle Utilizing Machine Learning to Cross-Check Traffic Data and Understand Urban Mobility		5. Report Date March 2021 6. Performing Organization Code N/A
7. Author(s) Zhao Zhang, Xianfeng Terry Yang		8. Performing Organization Report No. N/A
9. Performing Organization Name and Address University of Utah Department of Civil and Environmental Engineering 110 Central Campus Drive, Suite 2000 Salt Lake City, Utah 84092		10. Work Unit No. 5H08434H 11. Contract or Grant No. 20-8311
12. Sponsoring Agency Name and Address Utah Department of Transportation 4501 South 2700 West P.O. Box 148410 Salt Lake City, UT 84114-8410		13. Type of Report & Period Covered Final Sept 2019 to March 2021 14. Sponsoring Agency Code UT19.301

15. Supplementary Notes

Prepared in cooperation with the Utah Department of Transportation, the Utah Department of Technology Services, and the U.S. Department of Transportation, Federal Highway Administration

16. Abstract

At UDOT, PeMS stores point data collected by roadside radar sensors, loop detectors, and micro-loops, while ClearGuide contains statewide probe data. PeMS data can greatly support mobility pattern studies but are only available at detector locations. In contrast, ClearGuide can provide statewide traffic speed information based on probe vehicle data. Hence, the speed estimates in ClearGuide have a high potential to be biased due to the low probe penetration rate. Comparisons between PeMS and ClearGuide data reveal significant differences in 5-min speed, which further prove the data bias and inaccuracy in ClearGuide. However, as ClearGuide can provide statewide traffic information, it has been used to support many traffic operation tasks when PeMS data are not available. Thus, the lack of correcting ClearGuide data can result in unreliable inputs and consequently the failure of traffic operational activities. To tackle this issue, this research aims to develop a set of machine learning models to integrate these two data sources, mitigate data variations, and produce more reliable estimations of the statewide traffic patterns. The research objectives are achieved by two steps. The first step utilizes regression machine learning algorithms to estimate traffic state using probe vehicle and sensor detector data. Also, performance of those machine learning algorithms is compared using a novel estimation framework. The second step aims to develop a hybrid machine learning approach, by creating a new training variable based on the second-order traffic flow model, to improve the accuracy of traffic state estimation.

17. Key Words Traffic speed, machine learning, hybrid learning, probe data, stationary data.		18. Distribution Statement Not restricted. Available through: UDOT Research Division 4501 South 2700 West P.O. Box 148410 Salt Lake City, UT 84114-8410 www.udot.utah.gov/go/research		23. Registrant's Seal N/A
19. Security Classification (of this report) Unclassified	20. Security Classification (of this page) Unclassified	21. No. of Pages 58	22. Price N/A	

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

ANN	Artificial Neural Network
ARIMA	Auto Regressive Integrated Moving Average
ARZ	Aw-Rascle-Zhang
ATMS	Advanced Traffic Management System
ATIS	Advanced Traveler Information System
CFL	Courant-Friedrichs-Lewy
CTM	Cell Transmission Model
ES	Exponential Smoothing
GBDT	Gradient Boosting Decision Tree
GIS	Geographic Information System
ITS	Intelligent Transportation Systems
KF	Kalman Filter
LWR	Lighthill-Whitham-Richards
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
ML	Machine Learning
MLP	Multilayer Perceptron
PDE	Partial differential equations
PeMS	Performance Measurement System
PW	Payne-Whitham
RMSE	Root Mean Square Error
RF	Random Forest
SVM	Support Vector Machine
TFM	Traffic Flow Model
TOC	Traffic Operations Center
TSE	Traffic State Estimation
TSP	Traffic State Prediction
UDOT	Utah Department of Transportation
XGBoost	Extreme Gradient Boosting

EXECUTIVE SUMMARY

In UDOT's Traffic Operations Center (TOC), several online data platforms, such as Performance Measurement System (PeMS) and ClearGuide, are currently used for data visualization and sharing. PeMS stores point data collected by roadside radar sensors, loop detectors, and micro-loops, and ClearGuide contains statewide probe information. Comparing the data from PeMS and ClearGuide on the same freeway segment, this research has observed significant differences in 5-min speeds between them. Notably, PeMS data collected from road sensors are considered to be more accurate than ClearGuide data. Hence, the observed differences between PeMS and ClearGuide could indicate the high potential of data bias and inaccuracy in ClearGuide. However, as ClearGuide can provide statewide traffic information, it has been used to support many traffic operation tasks when PeMS data are not available. Hence, the lack of correcting ClearGuide data can result in unreliable inputs and consequently the failure of traffic operational activities.

To tackle this issue, this research aims to develop a set of machine learning (ML) models to integrate these two data sources, mitigate data variations, and produce more reliable estimations of statewide traffic speed and flow patterns. The research objectives are achieved by two primary steps. The first step utilizes regression ML algorithms to estimate traffic speed and flow based on probe vehicle and sensor detector data. Also, performances of selected ML algorithms are compared using a novel traffic estimation framework. Then, the one with the best performance can be identified. However, in some cases, the limitation of data quality remains a big challenge, where models developed in the first step may have downgraded performances. To overcome this problem, the second step aims to develop a hybrid ML approach, by creating a new training variable based on the second-order traffic flow model, to improve traffic state estimation. Comparisons between the hybrid approach and pure ML models show that the hybrid approach can effectively capture the time-varying pattern of the traffic and help improve the estimation accuracy.

1.0 INTRODUCTION

1.1 Problem Statement

During the last few decades, Intelligent Transportation Systems (ITS) have been widely deployed on freeway systems for improving traffic safety and efficiency, and offering better travel choices (e.g., departure time, route, mode, etc.) to travelers. The effectiveness of ITS depends on the quality of obtained traffic information, especially for Advanced Traffic Management Systems (ATMS) and the Advanced Traveler Information Systems [ATIS] (Ma et al., 2015). Hence, providing an accurate and timely traffic state (i.e., speed and flow) is critical to support the operation of ITS, which is also needed by individual travelers, business sectors, and transportation agencies. More specifically, traffic state not only can help transportation agencies find better countermeasures to mitigate traffic congestion and improve traffic safety but also can benefit travelers to preplan and reschedule trips (Lv et al., 2015; Ma et al., 2015; Wang et al., 2019). As shown in Figure 1.1, several UDOT online data platforms, including Performance Measurement System (PeMS) and ClearGuide, are currently used for data visualization and sharing. PeMS stores point data collected by roadside radar sensors, loop detectors, and micro-loops, and ClearGuide contains statewide probe data.

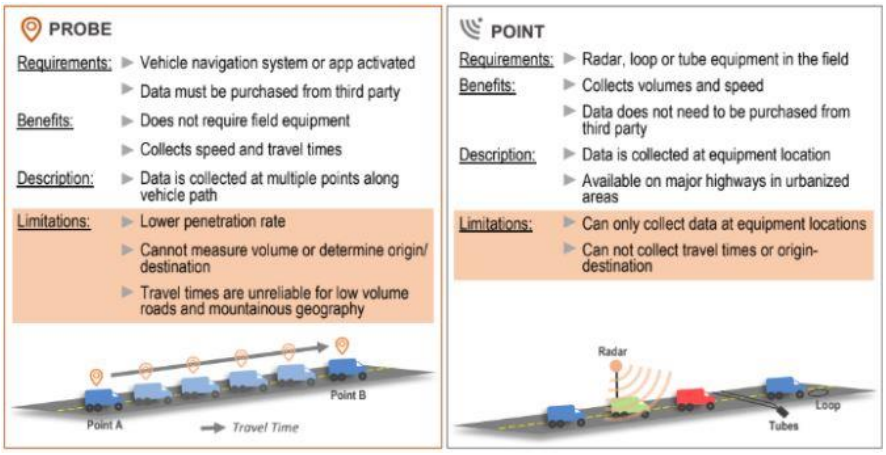


Figure 1.1 Two types of traffic data in UDOT

Point data in PeMS can greatly support freeway mobility pattern studies but are only available at the detector locations. In contrast, ClearGuide can provide statewide traffic speed

information based on the probe vehicle data. However, the estimated speeds in ClearGuide have a high potential to be biased due to the low probe penetration rate (around 2% of the entire traffic). Comparing the data from PeMS and ClearGuide on the same freeway segment, this research has observed significant differences in 5-min speed between them (see Figure 1.2). Notably, PeMS data collected from road sensors are considered to be more accurate than ClearGuide data. Hence, the observed differences between PeMS and ClearGuide could prove the data bias and inaccuracy in ClearGuide. However, as ClearGuide can provide statewide traffic information, it has been widely used by UDOT engineers to support many traffic operation tasks when PeMS data are not available. The lack of correcting ClearGuide data can result in unreliable inputs and consequently the failure of traffic operational activities.

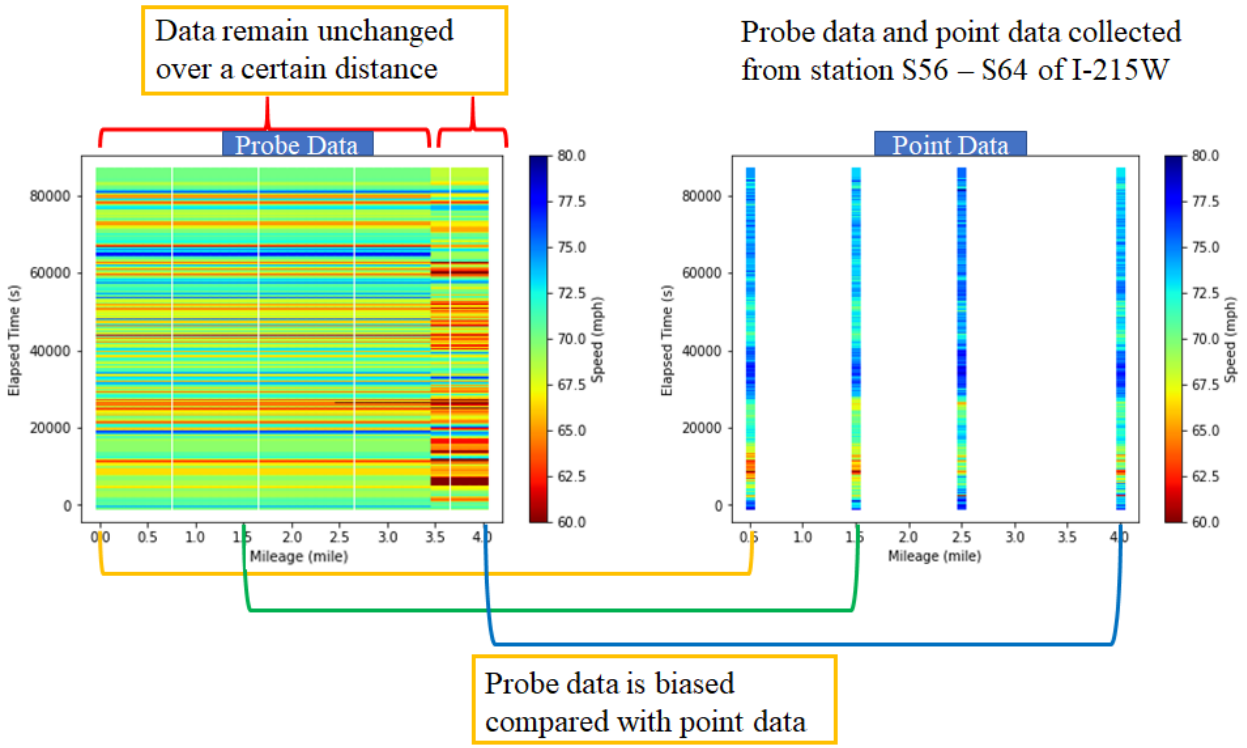


Figure 1.2 Parallel comparison of PeMS and ClearGuide speed

Recognizing the limitations in both PeMS and ClearGuide, this research aims to develop reliable models to integrate these two data sources, mitigate data variations, and produce more reliable estimations of statewide traffic patterns. In the literature, traffic state estimation (TSE)

has been recognized as an important tool by subject matter experts since the 1970s. A majority of existing studies on TSE were conducted by traditional traffic flow models based on detector measurements (i.e., flows and mean speed). These traditional models have limited ability to obtain accurate traffic states but can approximately estimate the traffic pattern in a wide range. Hence, using machine learning (ML) may become a better option.

By definition, ML refers to the study of algorithms that improve their performance "P" at some task "T" with experience "E". Figure 1.3 shows the main difference between ML models and traditional modeling approaches. The logic of traditional methods is to implement the developed model to process input data and then obtain the output (e.g., traffic flows). In contrast, ML would take both input data and expected output to conduct a training/learning process. Instead of being interested in acquiring specific outputs, ML will produce a trained model.

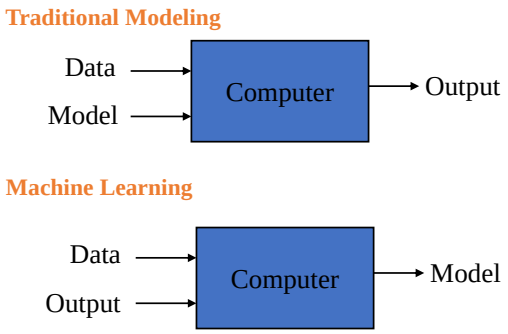


Figure 1.3 Difference between ML and traditional modeling

Compared with traditional modeling approaches, the benefits of ML are obvious. First, traditional methods are usually developed based on strong assumptions and can't accommodate data uncertainties, while ML is able to fully capture the stochastic natures of data. Second, ML can accommodate some difficult problems (e.g., voice recognition) that are even impossible to be modeled by traditional methods. Last but not least, a well-trained ML model can be implemented to generate the desired output in a much shorter time period because traditional methods have to go through the complicated solving process. However, ML also has its limitations. For example, ML is treated as a "black box", which makes its results hard to interpret with physical meanings. Also, due to its data-driven nature, the performance of ML highly relies on the quality and

quantity of training data. When the training dataset is either noisy or small, the performance of ML can be downgraded.

Leveraging ML for TSE, this research includes two primary steps. The first step utilizes regression ML algorithms to estimate traffic state based on both probe vehicle and detector data. Specifically, probe data and detector data are implemented as the input and output, respectively, of the ML training process. Also, performances of selected ML algorithms are compared using a novel estimation framework. The results show that the proposed framework can effectively capture time-varying traffic patterns and has a superior ability to accurately estimate traffic state in a timely manner. Using detector data as the benchmark, the comparison results show that the Random Forest (RF) model achieves the best performance in TSE.

The limitation of data quality remains a big challenge in some cases, where the model developed in the first step may have downgraded performance. To overcome this problem, the second step aims to develop a hybrid ML approach, by creating a new training variable based on the second-order traffic flow model, to improve the accuracy of TSE. Grounded on a novel integrated framework, the estimation is performed using hybrid ML techniques. All models are trained with the integrated dataset including the traffic flow model estimates, ClearGuide information, and PeMS data. The comparisons between the hybrid approach and pure ML models show that the hybrid approach can effectively capture the time-varying pattern of the traffic and help improve the estimation accuracy.

1.2 Objectives

Despite PeMS stationary data being more accurate, they are only available at the detection locations. In contrast, ClearGuide probe data can cover the entire freeway network, but may be biased. Therefore, the primary objective of this research project is to design a new modeling framework that can fuse both PeMS and ClearGuide data and develop ML models to estimate more accurate traffic states. More specifically, five classical ML models are tested and compared to explore the best TSE option. Expected results are shown in Figure 1.4. Considering that ML performance would be affected by the training data quality due to its data-driven nature, the secondary research objective is to develop a hybrid learning approach, grounded on the

macroscopic traffic flow model, to deal with the cases with data quality issues. When ML models are developed and validated, the tertiary research objective is to provide a plan that can help the UDOT Traffic Operations Center (TOC) to implement the developed models.

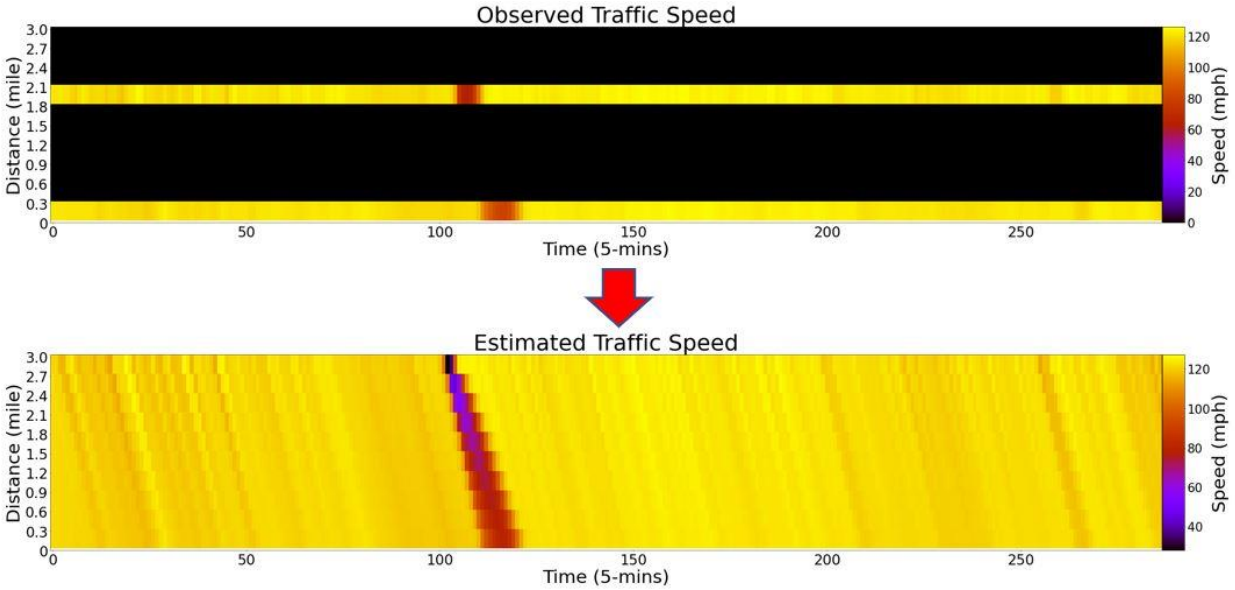


Figure 1.4 Objective of this research project

1.3 Scope

This project includes four main research tasks. In Task 1, traffic state data from both PeMS and ClearGuide platforms are obtained to conduct parallel comparisons. Notably, the obtained PeMS data is pre-corrected using the data screening algorithm developed in another research project titled "Multi-Stage Algorithm for Detection-Error Identification and Data Screening". In Task 2, five classical ML models are tested and their performances are compared. Then in Task 3, a hybrid ML modeling framework is introduced to further improve the ML performance. When all ML models are validated, a work plan is developed to discuss how to implement them into current UDOT data visualization platforms.

1.4 Outline of Report

This project report includes seven chapters and the outline is listed as follows:

- Introduction
- Literature Review
- Data Collection and Analysis
- Machine Learning for Traffic State Estimation
- Hybrid Machine Learning Model
- Recommendations and Implementation
- Conclusions

2.0 LITERATURE REVIEW

2.1 Overview

This chapter reviews studies related to TSE and traffic state prediction (TSP) in the literature, highlights existing research gaps, and identifies critical issues to be investigated. To facilitate the presentation, this chapter classifies existing studies into three categories: (1) TSE with traffic flow models; (2) pure ML for TSE and TSP; and (3) hybrid ML for TSE.

2.2 Traffic Speed Estimation with Traffic Flow Models

TSE is a method that can infer traffic state (e.g., flow, speed, density, etc.) using partially observed data from traffic sensors on the roadway system (Seo et al., 2017). Accurate and timely TSE is not trivial work due to the stochastic characteristics of the traffic. Previous studies have shown that the two most common TSE approaches on the freeway are model-driven approaches and data-driven approaches (Seo et al., 2017). These approaches are designed to simulate traffic dynamics, capture data noise, and estimate unobserved spatiotemporal traffic states. For model-driven approaches, in the early stages macroscopic traffic dynamics were found to be similar to hydrodynamics. By borrowing concepts from the fluid mechanism, flow, speed, and density were defined and their relationship, named the fundamental diagram, was discovered. Based on these definitions, macroscopic traffic flow models were developed based on the conservation law and momentum, and a set of kinematic wave models were also formulated (Seo et al., 2017). However, most models derived under ideal theoretical conditions require great efforts for parameter calibration and are challenging to work with noisy and fluctuating data collected by traffic sensors.

In general, model-driven approaches can be generally classified into three categories: (1) the first-order Lighthill-Whitham-Richards (LWR) model (Lighthill and Whitham, 1955; Richards, 1956), (2) the second-order Payne-Whitham (PW) model (Payne, 1971; Whitham, 1975), and (3) the second-order Aw-Rascle-Zhang (ARZ) model (Aw and Rascle, 2000; Zhang, 2002). The LWR model succeeds in mimicking simple traffic conditions (e.g., traffic jam and shockwave), but it cannot reproduce more complicated traffic phenomena well. Therefore, the

PW and ARZ models were developed by adding the momentum equations to capture complex traffic behavior. However, these models require tremendous computation efforts in some cases since they were derived under ideal theoretical conditions. To overcome the limitation of PW and ARZ models, partial differential equations (PDE) are utilized to discretize their model formulations by transferring the road segment and time interval into elements. In summary, the discrete reformulation can be categorized by: (a) the Godunov scheme (Lebacque, 1996); (b) the upwind scheme (Lebacque et al., 2007); (c) the Lax-Wendroff scheme (Michalopoulos et al., 1993); and (d) the Lax-Friedrichs scheme (Wong and Wong, 2002; Göttlich et al., 2013). The cell transmission model (CTM) is a simplified case of the Godunov scheme of the LWR discretized with the Courant-Friedrichs-Lewy (CFL) number equal to 1 (Daganzo, 1994). To extend the PW model, Papageorgiou et al. (1989) proposed a discrete PW-like TSE model named METANET, which can reproduce complex traffic phenomena and does not require tremendous computation efforts at a certain level. The METANET model has been successfully applied in many studies (Wang and Papageorgiou, 2005; Zhang et al., 2020).

2.3 Pure Machine Learning for TSE and TSP

In view of the increasing data availability, data-driven approaches such as ML models were developed for TSE because they do not require explicit theoretical assumptions and have a remarkably low computational cost in the testing phase. However, it is difficult to deduce the insights of data-driven approaches that can be considered as “black boxes” (Seo et al., 2017). In the literature, data-driven approaches include autoregressive integrated moving average [ARIMA] (Zhong et al., 2004), Bayesian network (Ni and Leonard, 2005), kernel regression (Yin et al., 2012), fuzzy c-means clustering (Tang et al., 2015), k-nearest neighbors clustering (Tak et al., 2016), stochastic principal component analysis (Li et al., 2013; Tan et al., 2014), Tucker decomposition (Tan et al., 2013), deep learning (Duan et al., 2016; Polson and Sokolov, 2017b; Wu et al., 2018), Bayesian particle filter (Polson and Sokolov, 2017a), etc. However, the performance of those models relies on the data quality due to the data-driven nature. The model performance may drop when (a) training data are scarce and insufficient to reveal the complexity of the system and (b) training data includes random noisy and incorrect information.

Following the same path, some other regression ML models were developed to handle the time-varying pattern of the traffic state and conduct TSP. For example, Zhang et al. (2019) applied a deep learning-based multitask learning model to forecast network traffic speed. Non-neural network ML approaches are also applicable for predicting traffic states. Such approaches include the RF (Hamner 2010; Leshem and Ritov 2007; Wang et al., 2016), Support Vector Machine [SVM] (Wang and Shi 2013), Gradient Boosting Decision Tree [GBDT] (Ding et al., 2016; Ma et al., 2017; Yang et al., 2017; Zhang and Haghani 2015), and Extreme Gradient Boosting (XGBoost) (Wang et al., 2016). These studies show that these ML techniques have an excellent ability to capture the stochastic characteristics of the traffic state, which motivate implementing them on both TSE and TSP.

The SVM-based method, which estimates the regression based on a series of kernel functions, has an ability to convert the lower-dimensional input data to a higher dimensional feature space via a nonlinear relationship. It then performs linear regression within this space (Smola and Schölkopf 2004). Time series and regression problems can be effectively modeled by SVM-based traffic models, which have been proven by several studies (Wu et al., 2004; Asif et al., 2014; Zhang and Liu 2009). The GBDT model, proposed by Friedman (Friedman 2002), is widely implemented for regression and classification problems. It combines the strengths of boosting algorithms and decision trees, which can effectively solve traffic-related time series and regression problems (Ding et al., 2016; Ma et al., 2017; Yang et al., 2017; Zhang and Haghani 2015).

The RF model, developed by Leo Breiman (2001), is an ensemble technique that can be performed to solve both regression and classification problems. Notably, although GBDT also combines a set of "weak" learners, the main difference between GBDT and RF is that the tree in GBDT fits the previous tree's residual. Hence, GBDT can reduce the bias while RF can reduce variance. The overfitting problem can be prevented most of the time by Breiman's "bagging" idea, which randomly selects features. The ability of RF to predict traffic speed was proved by several studies (Hamner 2010; Leshem and Ritov 2007). The XGBoost algorithm, proposed by Chen and Guestrin (2016), is an improved algorithm of Gradient Boosting. It develops a "strong" learner through additive training strategies by combining predictions of a set of "weak" learners, which can decrease variance by adding

regularization terms. It is widely implemented in linear regression, linear classification, and logistic regression problems. In the literature, Wang et al. (2016) applied the XGBoost algorithm to predict traffic flow on urban transportation networks.

Artificial Neural Network (ANN) is also considered as an effective method for traffic state prediction. This method can accommodate multi-dimensional data, flexible model structure, strong generalization, learning ability, and adaptability (Karlaftis and Vlahogianni 2011). Compared with statistical methods, ANN does not require underlying data assumptions and can effectively deal with missing and noisy inputs (Karlaftis and Vlahogianni, 2011). Many researchers showed that ANN could effectively predict traffic state [e.g., flow and speed] (Taylor and Meldrum, 1995; van Lint et al., 2005; Zeng et al., 2008; Kumar et al., 2013). In summary, these five types of ML models – SVM, GDBT, RF, XGBoost, and ANN – are proficient in dealing with regression problems and can effectively capture time-varying traffic patterns.

2.4 Hybrid Machine Learning for TSE

As shown above, model-driven and data-driven approaches are the two most common TSE methods on freeways (Seo et al., 2017). According to the existing studies, both approaches have their advantages and drawbacks. Model-driven approaches can simulate traffic dynamics and predict unobserved spatiotemporal traffic states with a limited amount of observed traffic information. The data-driven approaches are better at capturing the stochastic characteristics of traffic flow based on a massive amount of historical data. As proven by existing studies, the estimation methodology and the data quality are the two essential parts of TSE (Xiao et al., 2018). Hence, to overcome the limitations of both approach types, data expansion, data fusion, and hybrid approaches were developed for TSE. These hybrid concepts can combine the advantages of different data sources and different methods. Particularly, data expansion algorithms are included to supplement missing values in the traffic state data collected by roadside sensors (Lederman and Wynter, 2011).

The effectiveness of data fusion techniques for improving the accuracy of travel time estimation was proved by some studies [Anusha et al., (2012), Zhu et al., (2018)]. The hybrid data-driven and model-driven approaches for traffic time estimation and forecasting were

implemented and evaluated by previous studies (You and Kim, 2000; Yu et al., 2010; Hofleitner et al., 2012; Allström et al., 2016; Kumar et al., 2017; Sharmila et al., 2019). You and Kim (2000) proposed a hybrid nonparametric regression model with geographic information systems (GIS) information to forecast link travel times in congested road networks. Yu et al. (2010) proposed a hybrid model that combines SVM and Kalman filtering (KF) for effectively predicting bus arrival time. Hofleitner et al. (2012) presented a hybrid modeling framework that combines the advantages of pure statistical and traffic flow models to forecast travel time on local arterials. Allström et al. (2016) applied a hybrid approach that adopted ANN with output from the CTM for short-term traffic state and travel time prediction. Kumar et al. (2017) proposed an integrated method, which uses exponential smoothing (ES) and KF to estimate travel time as a new observation for ARIMA models, to estimate bus travel time. Sharmila et al. (2019) proposed a hybrid model that combines data-driven and model-driven approaches for corridor-level travel time estimation. These studies demonstrated that data expansion, data fusion, and hybrid approaches could improve the estimation and prediction accuracy of traffic measures.

Based on the literature and considering the advantages and drawbacks of both traffic flow and ML models, using the hybrid of the traffic flow model estimates and probe data as the input for ML models would potentially produce a better alternative to address the TSE challenges.

2.5 Summary

This chapter provided a comprehensive review of current research on the subject of TSE. Existing TSE methods can be generally classified into three categories: model-driven approaches, data-driven approaches, and hybrid approaches. Studies have demonstrated the effectiveness of proposed methods for TSE or TSP with multiple data sources. However, they may fall short of providing accurate TSE on freeways when sufficient and reliable training data are not available. Most existing hybrid approaches were proposed for travel time estimation and prediction on local arterials. The issue of applying hybrid approaches for TSE on freeways still lacks investigation.

3.0 DATA COLLECTION AND ANALYSIS

3.1 Overview

Data utilized in this study are obtained from UDOT PeMS and ClearGuide platforms, where PeMS provides stationary detector data and ClearGuide offers probe data. ClearGuide speed information is the average of probe vehicles' speeds. As aforementioned, probe vehicle data can provide statewide traffic information. However, such data have a relatively low resolution (e.g., traffic speed remains constant over a long freeway segment) and are often biased. To clearly demonstrate the data issues in ClearGuide and make comparisons between the two databases, this study first collects 14-day (1/7/2019 – 1/20/2019) data at the locations shown in Figure 3.1. More specifically, the collected data include PeMS traffic states from four stationary detectors (S56 – S64) and ClearGuide speed information between detectors S56 and S64.

Figure 3.1 Data collection locations

Table 3.1 summarizes the detailed description of collected data from both databases, which include speed, day of week, and time of day information from ClearGuide, and speed and flow information from PeMS.

Table 3.1 Summary of collected data

	Data source	Variable	Number of records
Input	ClearGuide	Speed (mph) Day of Week (1 – 7) Time of Day (5-min interval)	4032
Label **Value**	PeMS	Speed (mph) Flow (veh/5-min)	4032

3.2 Data Analysis and Pre-Processing

To study potential data issues in ClearGuide, Figure 3.2 presents the time-dependent distribution of traffic speed patterns from ClearGuide. It can be observed that the speed patterns are identical at stations 56-57. This observation indicates the low resolution of ClearGuide, where the speed remains constant over a freeway segment for a given time step.

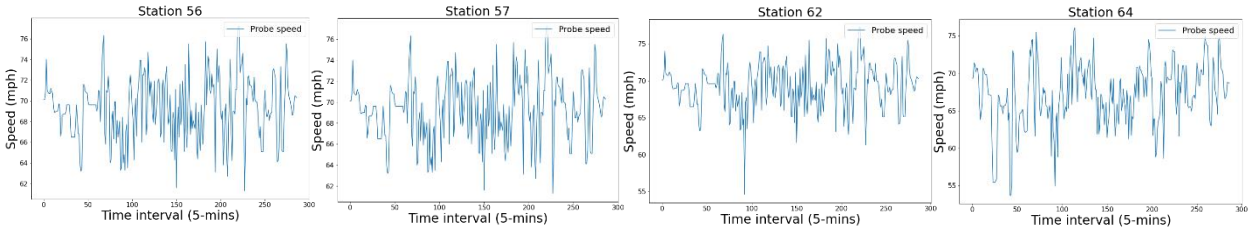


Figure 3.2 Probe speed data presentation

Moreover, considering ClearGuide data are provided by probe vehicles, which only represent a portion of the traffic, data bias issues may also exist. To prove that, Figure 3.3 illustrates the comparisons of speeds between ClearGuide and PeMS. As the PeMS data are already pre-corrected in this research using the data screening algorithm developed in another research project, they can be considered accurate and would serve as the "ground truth" for comparisons. Hence, differences between ClearGuide and PeMS data could confirm the existence of biased data in ClearGuide.

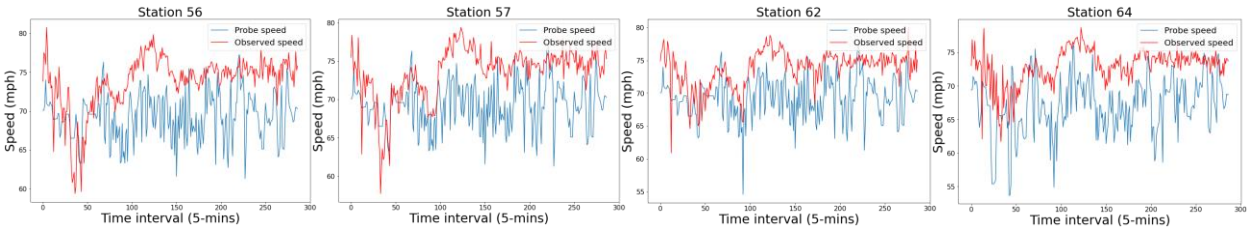


Figure 3.3 Comparison of probe and observed detector speed data

In the remainder of the report, pre-corrected data will be used to train ML models. In view of ML requirements, the raw speed and flow data were normalized, ranged from 0 to 1, using the following equation:

$$x_n = \frac{x_i - x_{min}}{x_{max} - x_{min}} \tag{3.1}$$

where x_n denote the normalized raw speed and flow data; x_i denote the raw speed and flow data; and x_{max} and x_{min} are the minimum and maximum of raw speed and flow data, respectively.

4.0 MACHINE LEARNING FOR TRAFFIC STATE ESTIMATION

4.1 Overview

Probe data is a sample of information collected from vehicle navigation systems, cell phone applications, and fleet vehicles. Such data are available on statewide freeway segments in Utah. However, they are biased due to the low probe penetration rate. Point data, which were collected at single points by roadside stationary detectors (e.g., radar, loops, micro-loops, etc.), can offer more accurate traffic information. However, they are only available at sparse locations with detectors installed. Recognizing the limitation of both data types, this chapter presents a reliable freeway TSE framework, established by various regression ML models, to estimate accurate traffic speeds and flows. The remainder of this chapter is organized as follows: Section 4.2 introduces the ML-based TSE models, Section 4.3 offers an overview of the regression ML algorithms, Section 4.4 presents the results of numerical tests on the field data, and Section 4.5 summarizes the findings.

4.2 Pure Machine Learning for TSE

The proposed TSE modeling architecture is shown in Figure 4.1. After obtaining the traffic data, ML models are trained with grouped probe and observed (detector) data. Then, trained models are used to estimate the traffic state at segments without observed data, using probe data as input. The procedure for implementing the proposed estimation system can be stated as follows:

- *Step 1*: For a freeway segment i with probe data only, select the upstream and downstream stations that have both probe and observed data available for model training;
- *Step 2*: Train ML models (i.e., SVM, RF, GBDT, XGBoost, and ANN) at both upstream and downstream stations, based on the grouped dataset.

$$S_{\text{grouped}} = \{(x_1, y_1), (x_1, y_1), \dots, (x_n, y_n)\}$$

where x_i is the input that consists of probe speed, flow, location, and time of data; y_i is the label output that consists of the observed speed and flow by detectors; and n is the number of training samples;

- *Step 3*: Use the two trained models to estimate the traffic state for freeway segment i and select the one with better performance (e.g., lower RMSE); and
- *Step 4*: Repeat the process in steps 1-3 until all freeway segments without observed data are studied.

To illustrate the proposed modeling framework with case studies, one-day (1/7/2019) traffic information from three stations (S56, S57, and S64) is selected for TSE. Notably, this research assumes that accurate traffic information is only available at the locations with traffic sensors installed, as shown in Figure 4.1, while probe data are available over the entire segment.

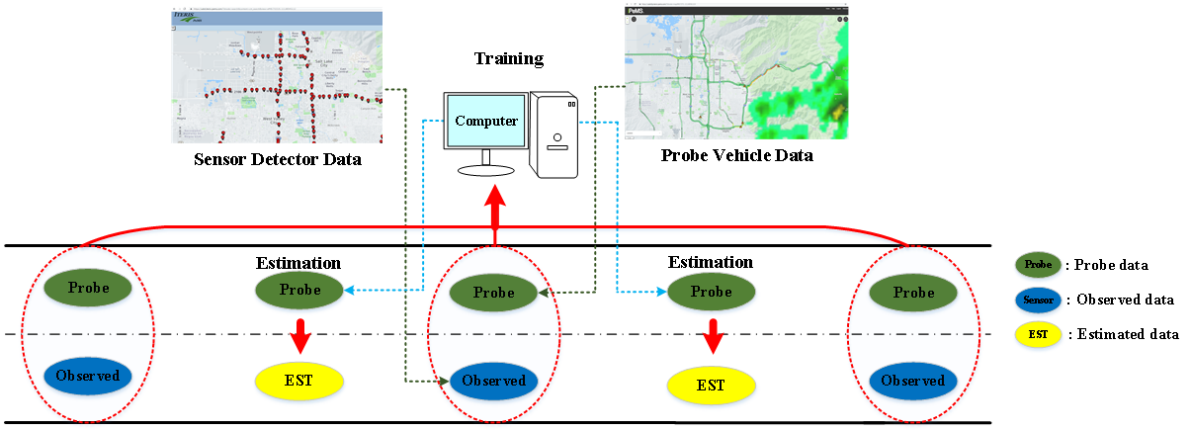


Figure 4.1 Architecture of proposed TSE modeling framework

4.3 Regression Machine Learning Algorithms Overview

4.3.1 Support Vector Machine (SVM)

SVM is a supervised artificial intelligence approach developed by Vapnik (2013). It is considered as an effective and efficient algorithm for regression and forecasting. The approximated function in the SVM algorithm can be depicted as follows:

$$f(x) = \omega\varphi(x) + b \tag{4.1}$$

where, $\varphi(x)$ is the higher-dimensional feature space; ω is the weights vector; and b is threshold. ω and b can be estimated by minimizing the following regularized risk function:

$$R(C) = C\frac{1}{n}\sum_{i=1}^{n} L(d_i, y_i) + \frac{1}{2}\|\omega\|^2 \quad (4.2)$$

where, $\frac{1}{2}\|\omega\|^2$ is the so-called regularization term; C is the penalty parameter of the error; d_i is the desired value; n is the number of observations; and $C\frac{1}{n}\sum_{i=1}^{n} L(d_i, y_i)$ is the empirical error. Then the function L_ε can be determined as below:

$$L_\varepsilon = |d - y| - \varepsilon |d - y| \geq \varepsilon \; or \; 0 \; otherwise \quad (4.3)$$

where ε is the tube size.

The approximated function in Equation 4.1 can be rewritten by introducing Lagrange multipliers and exploiting the optimality constraints:

$$f(x, \alpha_i, \alpha_i^*) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) K(x, x_i) + b \quad (4.4)$$

where, $K(x, x_i)$ is the kernel function. A more detailed description of the computational procedure of SVM can be found in Vapnik (Vapnik, 2013).

4.3.2 Random Forest (RF)

The RF model, also called random decision forests model, is an ensemble technique that can be performed in both regression and classification problems (Cutler et al., 2012). Assuming that *x* is a set of explanatory variables (i.e., speed, flow, estimated speed, and time in this study) and *F*(*x*) is the response variable y (i.e, speed and flow), the training set can be described as $x = x_1, x_2, \ldots, x_n$ with response to $Y = y_1, y_2, \ldots, y_n$. The algorithmic structure of RF is shown in Figure 4.2 and the algorithm can be summarized as three steps:

Step 1: Randomly select *n* subsamples;

Step 2: Train regression tree for each sample;

Step 3: Average all prediction results from all trees.

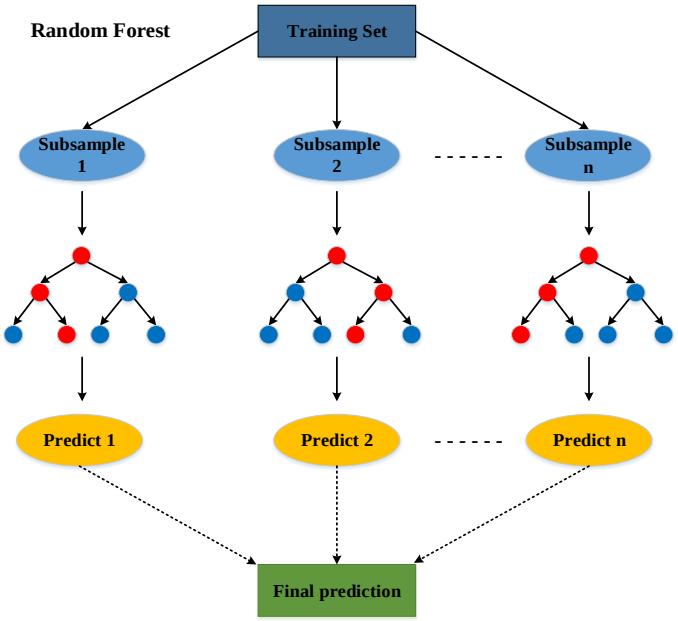


Figure 4.2 Modeling framework of Random Forest

4.3.3 Gradient Boosting Decision Tree (GBDT)

GBDT, proposed by Friedman (Friedman 2002), combines the strengths of boosting algorithms and decision trees. It has been widely implemented in regression and classification problems. Denoting $\{x_i, y_i\}_{i=1}^{n}$ as the training dataset, *h(x)* as the basic learner where $x_i = (x_{1i}, x_{2i}, \dots, x_{pi})$, p as the number of predicted variables, and y_i as the predicted label, the GBDT algorithm can be expressed as follows:

Step 1: obtain the initial constant $f_0(x)$:

$$f_0(x) = \arg\min_{\beta} \sum_{i=1}^{N} L(y_i, \beta) \tag{4.5}$$

Step 2: for m = 1: M (M is the times of iteration), compute the negative gradient using the following equation:

$$y_i{}^{*} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{f(x)=f_{m-1}(x)}, \quad i = \{1, 2, \dots, N\} \tag{4.6}$$

Step 3: fit the sample data and obtain the initial model by basic classifiers based on the least square approach, obtain parameter α_m, and fit the model $h(x_i; \alpha_m)$ by the following equation:

$$\alpha_m = \arg\min_{\alpha,\beta} \sum_{i=1}^{N} [y_i{}^* - \beta h(x_i;\alpha)]^2 \quad (4.7)$$

Step 4: compute the new gradient step size:

$$\beta_m = \arg\min_{\alpha,\beta} \sum_{i=1}^{N} L(y_i, F_{m-1}(x) + \beta h(x_i;\alpha)) \quad (4.8)$$

Step 5: update the model as follows:

$$f_m(x) = f_{m-1}(x) + \beta_m h(x_i;\alpha) \quad (4.9)$$

More detailed information about the GBDT model can be found in (Friedman 2002).

4.3.4 Extreme Gradient Boosting (XGBoost)

XGBoost, proposed by Chen and Guestrin (2016), is an improved algorithm of Gradient Boosting and is widely implemented in linear regression, linear classification, and logistic regression problems. The general prediction function at step t is depicted as follows:

$$f_i{}^{(t)} = \sum_{k=1}^{t} f_k(x_i) = f_i{}^{(t-1)} + f_t(x_i) \quad (4.10)$$

where, $f_t(x_i)$ is the learner at step t; $f_i{}^{(t)}$ and $f_i{}^{(t-1)}$ are the predictions at steps t and $t-1$; and x_i is the input variable.

To prevent the problem of over-fitting, the XGBoost model evaluates the "goodness" of model from the original function that is as shown below:

$$Obj^{(t)} = \sum_{k=1}^{n} l(\overline{y_i}, y_i) + \sum_{k=1}^{n} \Omega(f_i) \quad (4.11)$$

where, l is the loss function; n is the number of observations; and Ω is the regularization term which is defined as:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda||\omega||^2 \quad (4.12)$$

where, λ is the regularization parameter; γ is the minimum loss needed to further partition the leaf node; and ω is the vector of scores in the leaves. More detailed information regarding computation procedures of XGBoost can be found in (Chen and Guestrin 2016).

4.3.5 Artificial Neural Network (ANN)

ANN is a popular artificial intelligence approach that has been widely implemented in a variety of transportation problems. In the literature, most implemented ANN models are multilayer perceptron (MLP) models that can be expressed as follows:

$$y = h\left(\varphi_0 + \sum_{j=1}^{N} \varphi_j \, g\left(\sum_{i=1}^{M} \theta_i \, x_i\right)\right) \tag{4.13}$$

where, M and N denote the number of neurons in the input layer and hidden layer, respectively; g and h represent the transfer functions for the input layer and hidden layer; and the vector matrices of θ and φ denote the weight values for neurons in both the input layer and hidden layer, respectively.

4.4 Numerical Test

4.4.1 Model Performance Measurement

To evaluate the performance of different ML models in TSP, this research selects three common statistical indicators: Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and Mean Absolute Error (MAE), which are defined in Equations 4.14 - 4.16:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i\,*)^2} \tag{4.14}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i\,* - y_i|}{y_i} * 100\% \tag{4.15}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y} - y_i| \tag{4.16}$$

where, y_i is the observed traffic speed and flow and $y_i\,*$ is the estimated traffic speed and flow.

4.4.2 Case Setting

A stretch of interstate freeway I-215 (mileposts 16.17 – 18.7) in Salt Lake County, Utah, was selected to evaluate the effectiveness of the proposed TSE system. As shown in Figure 4.3, the observed data are available at the detection stations, indicated by blue and red icons, and the probe data can be retrieved over the entire segment.

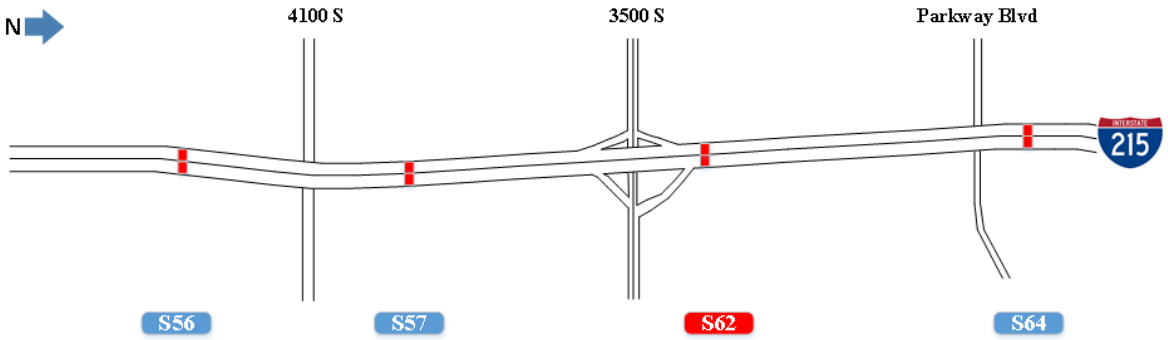


Figure 4.3 Overview of the study site

Two modeling frameworks with different input were analyzed to evaluate the effectiveness of the proposed TSE system: 1) pure ML (termed as pure-ML) that utilizes spatiotemporal information as the input data and the observed traffic state as labeled output; and 2) pure ML with probe data (termed as pure-ML with probe) that employs spatiotemporal information and probe data as input and the observed traffic state as labeled output. The pure-ML is a common scenario in TSE problems, which could be used to test TSE performance when only observed detector data are available. The pure-ML with probe data is used to evaluate TSE performance when additional probe data are further adopted as training variables. Two-week (1/7/2019 – 1/20/2019) observed and probe data from S56, S57, and S64 were grouped to train pure-ML models and all trained models were tested at the location of S62. Table 4.1 summarizes the details of input, output, and label for the ML models. All ML algorithms are implemented in Python with library scikit-learn (Pedregosa et al., 2011) and are tuned by the grid search approach.

Table 4.1 Summary of training variables in both types of ML

Cases	Data	Variables
Pure-ML	**Input**	Time (5-min interval) Distance (miles)
	Output	Speed (mph) Flow (veh/5-min)
	Label	Observed speed (mph) Observed flow (veh/5-min)
Pure-ML with probe	**Input**	Time (5-min interval) Distance (miles) Probe speed (mph)
	Output	Speed (mph) Flow (veh/5-min)
	Label	Observed speed (mph) Observed flow (veh/5-min)

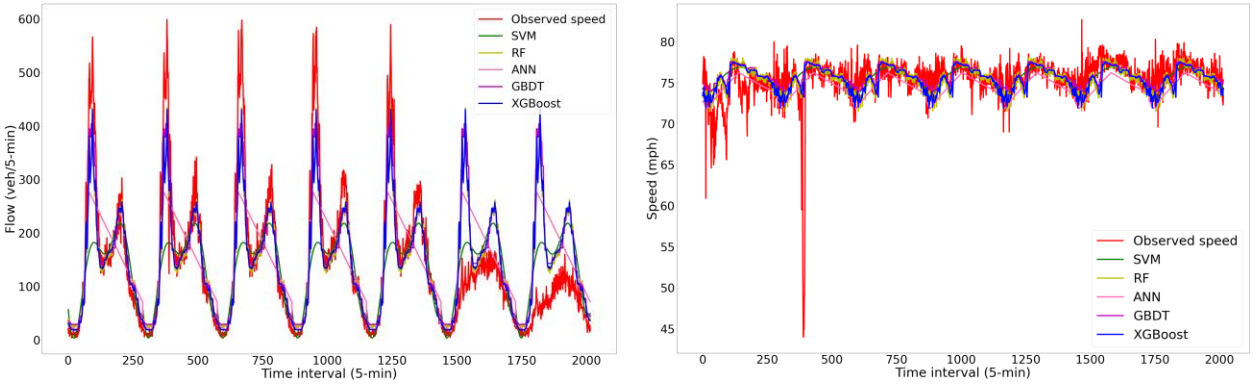
4.4.3 Results Analysis and Comparison

Table 4.2 shows the TSE results from all five pure-ML models. The resulting flow RMSEs are over 69 veh/5-min, the MAPEs are over 50%, and the MAEs are over 41 veh/5-min. Also, the speed RMSEs are over 3.35 mph, the MAPEs are over 2.60%, and the MAEs are over 1.80 mph. All three performance indicators for flow estimation are relatively high and thus are not acceptable. Although all pure-ML models provide relatively low performance, they still have the potential to improve speed estimation accuracy. Figure 4.4 compares the estimated flow and speed to the observed data, which indicates all five pure-ML models are not able to accurately estimate speed and flow. To further confirm this finding, the most accurate estimations by the pure-XGBoost were compared to the observed speed and flow. In Figure 4.5, if the coefficient of the trend line is close to one and the intercept is close to zero, the estimation results will be considered to be similar to the ground truth. In this case, the coefficient is 0.64 and the intercept is 55.13 for flow, and the coefficient is 0.13 and the intercept is 65.68 for speed.

Table 4.2 Estimation results of pure-ML models

Models	Flow RMSE	Flow MAPE	Flow MAE	Speed RMSE	Speed MAPE	Speed MAE
Pure-SVM	93.03	51.81%	54.07	3.46	2.67%	1.84
Pure-RF	69.87	59.88%	43.51	3.39	2.72%	1.89
Pure-ANN	85.70	67.55%	59.68	3.48	3.11%	2.21
Pure-GBDT	70.40	64.97%	42.02	3.47	2.62%	1.80
Pure-XGBoost	69.43	50.18%	41.81	3.39	2.69%	1.87

* Flow unit: (veh/5-min); Speed unit: mph



(a) Flow of pure-ML (b) Speed of pure-ML

Figure 4.4 Pure-ML estimates vs. ground truth

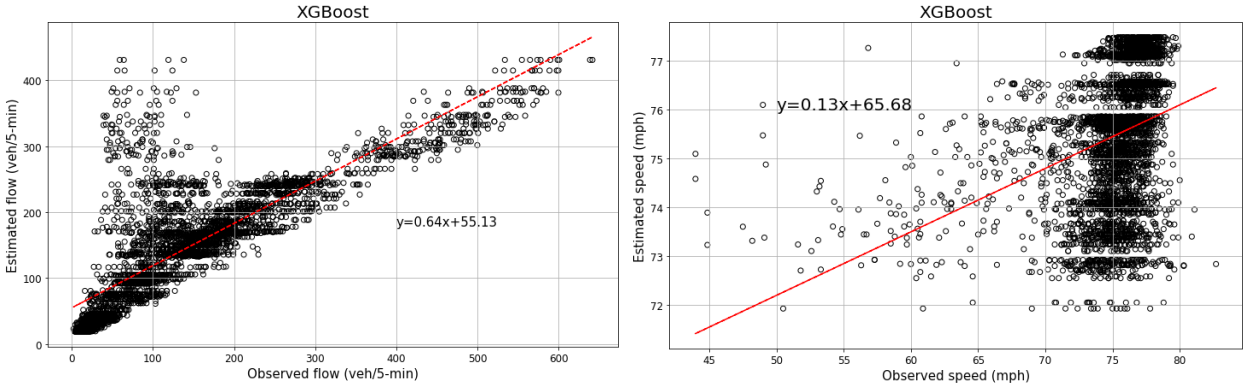


Figure 4.5 Comparison between pure-XGBoost estimates and ground truth

Table 4.3 demonstrates the bias of the original probe data and shows the TSE results of different pure-ML models with probe data. The probe data yields 5.57 mph for RMSE, 6.18% for MAPE, and 7.42 mph for MAE for speed estimation. Estimated speeds are biased compared to the ground truth. However, they might be valid additional inputs for pure-ML models to improve TSE accuracy because most probe vehicles are fleet vehicles that usually drive at lower speeds.

To test this hypothesis, all five pure-ML models are implemented by using the probe data as additional inputs. The highest flow RMSE, MAPE, and MAE are 36.26 veh/5-min, 35.16%, and 26.28 veh/5-min, respectively, and the highest speed RMSE, MAPE, and MAE are 3.00 mph, 3.03%, and 2.20 mph, respectively. Such results indicate that TSE by pure-ML models with probe data are at an acceptable level and TSE accuracy is dramatically improved compared to that from the pure-ML models. Table 4.4 displays the percentage of model performance improvement for the five pure-ML models with probe compared to the pure-ML models. The algorithm with the highest improvement is bolded. The pure-RF with probe has the highest improvements for all categories except flow MAPE. It indicates that TSE by pure-RF can be significantly enhanced with the addition of probe data.

Table 4.3 Estimation results of pure-ML models with probe

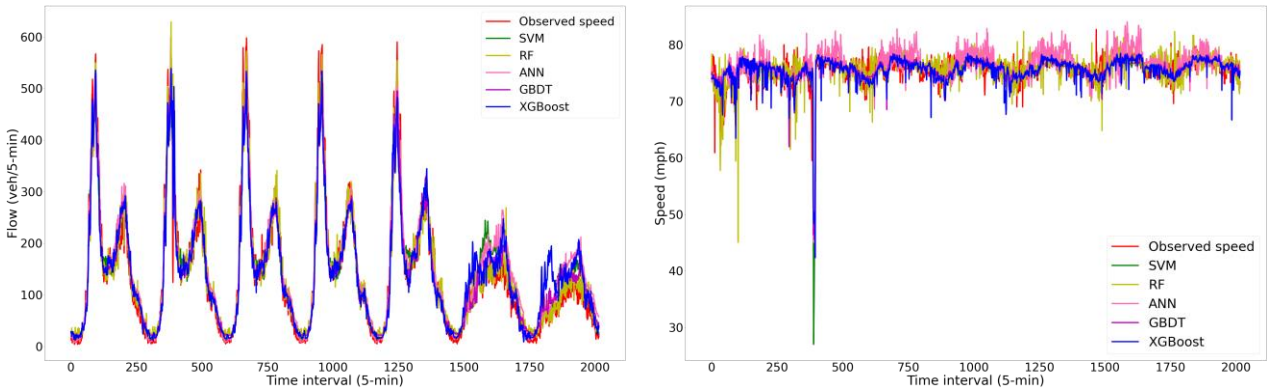
Method	Flow RMSE	Flow MAPE	Flow MAE	Speed RMSE	Speed MAPE	Speed MAE
Probe data	N/A	N/A	N/A	5.57	6.18%	7.42
Pure-SVM with probe	35.83	30.26%	24.89	2.58	2.20%	1.56
Pure-RF with probe	26.17	35.11%	18.69	2.23	1.99%	1.45
Pure-ANN with probe	36.26	31.87%	26.28	3.00	3.03%	2.20
Pure-GBDT with probe	29.47	34.86%	21.48	2.33	2.04%	1.46
Pure-XGBoost with probe	35.27	35.16%	25.14	2.43	2.16%	1.55

* Flow unit: (veh/5-min); Speed unit: mph

Table 4.4 Model performance improvement by additional probe data

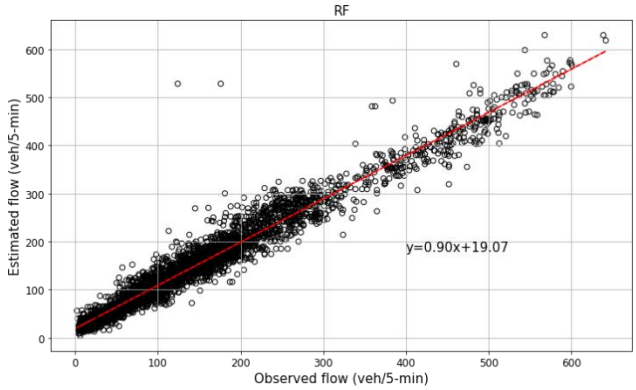
Method	Flow RMSE	Flow MAPE	Flow MAE	Speed RMSE	Speed MAPE	Speed MAE
Pure-SVM with probe	61.49%	41.59%	53.97%	25.43%	17.60%	15.22%
Pure-RF with probe	**62.54%**	41.37%	**57.04%**	**34.22%**	**26.84%**	**23.28%**
Pure-ANN with probe	57.69%	**52.82%**	55.97%	13.79%	2.57%	0.45%
Pure-GBDT with probe	58.14%	46.34%	48.88%	32.85%	22.14%	18.89%
Pure-XGBoost with probe	49.20%	29.93%	39.87%	28.32%	19.70%	17.11%

Figure 4.6 shows the estimated flow and speed by the five pure-ML models with probe data compared to the observed traffic state, which indicates the good performances of them in TSE. To further confirm this finding, the most accurate estimations by pure-RF with probe are selected to compare with the observed speed and flow. As shown by the trend lines in Figure 4.7, the coefficient is 0.90 and the intercept is 19.07 for flow, and the coefficient is 0.81 and the intercept is 14.69 for speed. Compared with the trend lines in Figure 4.5, Figure 4.7 confirms the benefit of including probe vehicle data in the training process.

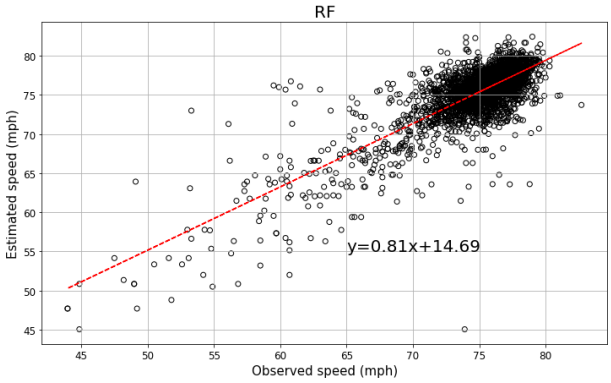


(a) Flow of pure-ML models with probe (b) Speed of pure-ML models with probe

Figure 4.6 Pure-ML with probe estimates vs. ground truth



(a) Flow of pure-RF with probe



(b) Speed of pure-RF with probe

Figure 4.7 Comparison between TSE by pure-RF with probe data and ground truth

4.5 Summary

ML techniques have a superior ability to capture the stochastic characteristics of traffic and estimate traffic speed accurately. However, clear guidance on which types of models should be selected for specific TSE applications and how to further integrate the probe data for training is not well studied. This chapter provided a novel pure-ML TSE system, enabled by five regression ML models (i.e., SVM, RF, ANN, GBDT, and XGBoost) to estimate the traffic state based on both detector data and probe data. To evaluate the effectiveness of the proposed method, the TSE system was implemented on I-215 in Salt Lake County, Utah. Based on the numerical results, the performance of all ML models could be dramatically improved by adding probe data as additional training variables. This is a pioneering study that applied regression ML techniques to freeway traffic estimation using both station-based and GPS-based data. The research findings also indicate that statewide deployment of probe data systems on freeways offers the possibility of adopting ML techniques to improve freeway management.

5.0 HYBRID MACHINE LEARNING MODEL

5.1 Summary

As shown in Chapter 4, adding probe vehicle data into the training process can help improve the ML TSE performance. However, in some cases, probe data are either not available or contain large amounts of noise, which can greatly affect the estimation accuracy. Such data issues commonly exist in practice and often bring difficulties in developing reliable and accurate ML. To tackle this issue, this chapter introduces a new hybrid ML framework that integrates the second-order macroscopic traffic flow model with ML. The remainder of this chapter is organized as follows: Section 5.2 introduces the hybrid ML approach for TSE, Section 5.3 presents numerical test results, and Section 5.4 summarizes the key findings.

5.2 Hybrid Machine Learning Model for TSE

5.2.1 Second-Order Macroscopic Traffic Flow Model

In this research, a well-calibrated second-order macroscopic traffic flow model developed by Papageorgiou et al. (1990) is employed to estimate the freeway traffic state. As shown in Figure 5.1, the target freeway segment is conceptually divided into N subsegments with a unit length of ΔL (e.g., 0.5 miles). For each subsegment i, the mean density, $d_j(k)$, can be determined by the difference between the input and output flows as follows:

$$d_i(k+1) = d_i(k) + \frac{\Delta T}{\lambda_i \Delta L}[q_{i-1}(k) - q_i(k) + r_i(k) - s_i(k)] \tag{5.1}$$

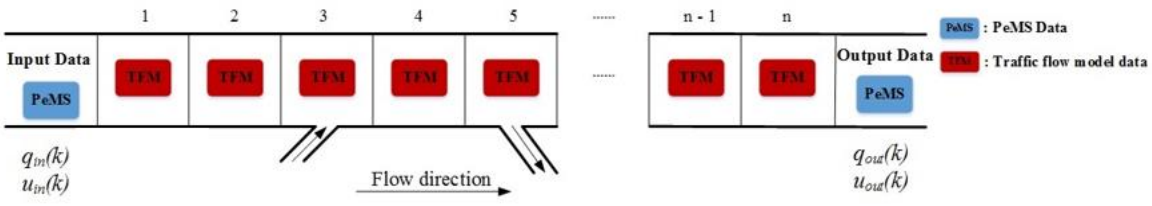


Figure 5.1 Freeway segmentation in the traffic flow model

In Eq. (5.1), $r_i(k)$ is the on-ramp flow rate entering subsegment i during interval k; $s_i(k)$ is the off-ramp flow rate leaving subsegment i during interval k; and $d_i(k)$ is the mean traffic density per lane in the subsegment i during interval k. To dynamically update the average speed, $u_i(k)$, a closed-form equation developed by the METANET model (Papageorgiou et al., 1990), is adopted:

$$u_i(k+1) = u_i(k) + \frac{\Delta T}{\tau_i}[V_i\{d_i(k)\} - u_i(k)] + \frac{\Delta T}{L_i}u_i(k)[u_{i-1}(k) - u_i(k)] - \frac{\gamma_i \Delta T}{\tau_i \Delta L}\frac{[d_{i+1}(k) - d_i(k)]}{[d_i(k) + \kappa]} \quad (5.2)$$

where $V[d_i(k)]$ is the static speed for segment i at time k given the density $d_i(k)$:

$$V[d_i(k)] = u_f \exp\left[-\frac{1}{a}\left(\frac{d_i(k)}{d_{cr}}\right)^a\right] \quad (5.3)$$

$u_i(k)$ is the mean speed in subsegment i during interval k; γ, τ, κ and a are traffic state model parameters; ΔL is the length of each freeway subsegment; and λ_i is the number of lanes in subsegment i. Also, the relationship between flow, density, and speed is given by the following:

$$q_i(k) = d_i(k)u_i(k)\lambda_i \quad (5.4)$$

where, i is the index of sub-sections of a freeway subsegment; k is the index of the time intervals; and $q_i(k)$ is the transition flow rate entering subsegment $(i+1)$ from i during interval k.

Using the observed flow and speed at upstream and downstream stations, on-ramps, and off-ramps, one can directly use Equations 5.1 - 5.4 to estimate the traffic speed evolution on the target freeway section.

5.2.2 Hybrid Machine Learning for TSE

As shown in Figure 5.2, the hybrid TSE model is constructed by integrating ML algorithms and macroscopic traffic flow models, based on both probe and detector data. The macroscopic traffic flow model (TFM) is first implemented to estimate the traffic state of the

selected freeway corridor based on the upstream and downstream detector data. ML models are then trained with the fused data that include observed detector data, TFM estimates, and probe data (if available). The trained models are used to estimate the traffic state at those locations without traffic detectors installed.

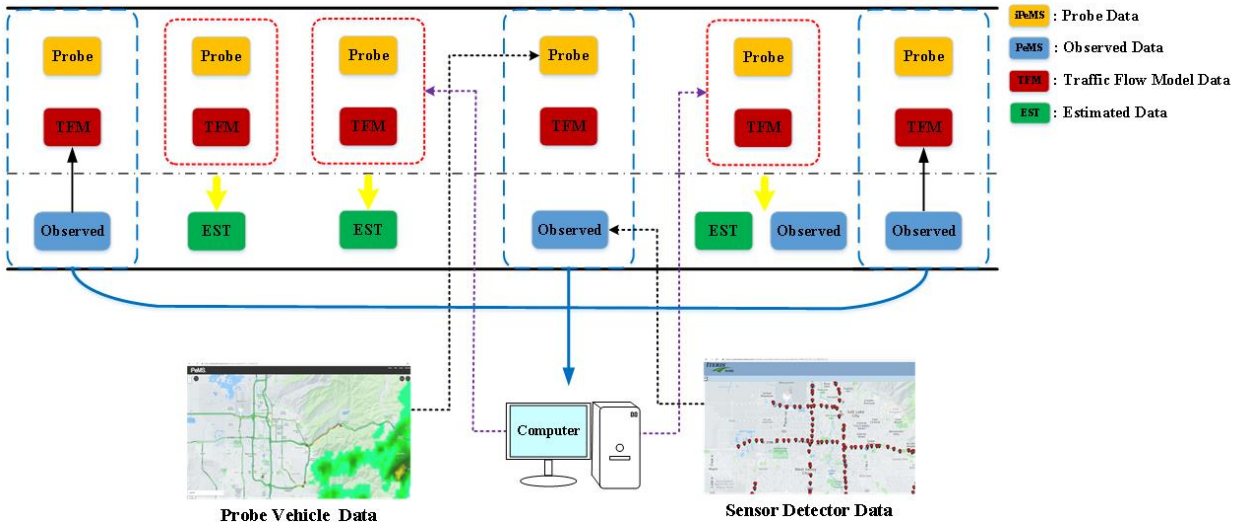


Figure 5.2 Architecture of hybrid machine learning model

After the model performance is validated, the proposed TSE system is implemented by the following steps:

- *Step 1*: Set the length of each freeway subsegment = 0.5 miles.
- *Step 2*: Select a set of freeway subsegments with both upstream and downstream detection stations and run the TFM to produce traffic state estimates for each subsegment between these two stations.
- *Step 3*: Train the machine leaning models (i.e., SVM, RF, GBDT, XGBoost, and ANN) with grouped dataset that includes both TFM estimates and observed data:

$$S_{\text{grouped}} = \{(x_1, y_1), (x_1, y_1), \dots, (x_n, y_n)\}$$

where x_i is the input of the training sample that consists of spatiotemporal information, TFM data, etc; y_i is the label value that consists of the traffic flow and speed of observed data; and n is the number of training samples.

- *Step 4*: Use the trained models to estimate the traffic state for each freeway subsegment i.
- *Step 5*: Repeat the process in steps 1-4 until all freeway subsegments without observed data are studied.

5.3 Numerical Test

5.3.1 Case Setting

In this chapter, data from the same freeway segment as the one used in Chapter 4 are obtained to evaluate the effectiveness of the proposed hybrid ML models. In this research, two cases with different datasets are analyzed for model evaluations:

1) Hybrid ML (termed as hybrid-ML) that utilizes spatiotemporal information and TFM data as input and treats the observed traffic state as the label; and

2) Hybrid ML with probe data (termed as hybrid-ML with probe) that uses spatiotemporal information, TFM data, and probe data as input, and treats the observed traffic state as label.

Because probe data are not commonly available in many states across the U.S., the first motivation of developing the hybrid-ML is to determine whether TFM estimates could be a replacement of probe data in the TSE. Then, the hybrid-ML with probe is built to test whether TSE accuracy could be further improved when probe data is also available. Two-week (1/7/2019 – 1/20/2019) detector data and TFM estimates from S56, S57, and S64 are grouped for training the hybrid-ML models, and the trained model is tested on S62. Table 5.1 summarizes the details of input, output, and label for the ML models. The calibrated METANET model parameters are listed in Table 5.2.

Table 5.1 Training data for hybrid-ML and hybrid-ML with probe

Cases	Data	Variables
	Input	Time (5-min interval) Distance (miles) TFM speed (mph)

hybrid-ML		TFM flow (vehicles/5-min)
	Output	Speed (mph) Flow (vehicles/5-min)
	Label	Observed speed (mph) Observed flow (vehicles/5-min)
hybrid-ML with probe	Input	Time (5-min interval) Distance (miles) TFM speed (mph) TFM flow (vehicles/5-min) Probe speed (mph)
	Output	Speed (mph) Flow (vehicles/5-min)
	Label	Observed speed (mph) Observed flow (vehicles/5-min)

Table 5.2 Initial parameters of the traffic flow model

Parameter	Value
n	9
λ_i	4
ΔT	1/360 (h)
u_f	75 (mi/h)
γ	20 (mi^2/h)
ΔL	0.5 (mi)
τ	0.05 (h)
α	1.4324
d_{cr}	59.30 (veh/mi)
κ	21 (veh/mi)

5.3.2 Results, Analysis, and Comparison

As shown in Table 5.3, the TFM yields a 56.81 veh/5-min RMSE, a 24.53% MAPE and a 32.34 veh/5-min MAE for flow, and a 2.81 mph RMSE, a 2.56% MAPE, and a 2.59 mph MAE for speed. The TFM can produce relatively low RMSE, MAPE, and MAE for both flow and speed estimates. Such finding demonstrates that the TFM could be a potential replacement of probe data when developing ML models for TSE. To test this hypothesis, five hybrid-ML models are implemented and the corresponding results are also presented in Table 5.3. Among all

hybrid-ML models without probe data, the lowest RMSE, MAPE, and MAE are 27.12 veh/5-min, 24.69%, and 19.69 veh/5-min, respectively, for flow estimation, and 2.20 mph, 1.87%, and 1.33 mph, respectively, for speed estimation.

Table 5.3 Comparison of different models

Method	Flow RMSE (veh/5-min)	Flow MAPE	Flow MAE (veh/5-min)	Speed RMSE (miles/h)	Speed MAPE	Speed MAE (miles/h)
Probe data	N/A	N/A	N/A	5.57	6.18%	7.42
TFM data	56.81	24.53%	32.34	2.81	2.56%	2.59
Pure-SVM	93.03	51.81%	54.07	3.46	2.67%	1.84
Pure-SVM with probe	35.83	30.26%	24.89	2.58	2.20%	1.56
Hybrid-SVM	42.91	24.69%	24.60	2.55	1.94%	1.37
Hybrid-SVM with probe	31.05	25.04%	20.17	2.00	1.72%	1.24
Pure-RF	69.87	59.88%	43.51	3.39	2.72%	1.89
Pure-RF with probe	26.17	35.11%	18.69	2.23	1.99%	1.45
Hybrid-RF	27.12	35.81%	19.69	2.20	1.87%	1.35
Hybrid-RF with probe	23.12	34.10%	17.62	1.69	1.62%	1.18
Pure-ANN	85.70	67.55%	59.68	3.48	3.11%	2.21
Pure-ANN with probe	36.26	31.87%	26.28	3.00	3.03%	2.20
Hybrid-ANN	35.60	37.95%	23.30	2.84	2.47%	1.79
Hybrid-ANN with probe	28.02	32.84%	19.87	2.17	2.22%	1.65
Pure-GBDT	70.40	64.97%	42.02	3.47	2.62%	1.80
Pure-GBDT with probe	29.47	34.86%	21.48	2.33	2.04%	1.46
Hybrid-GBDT	32.62	35.07%	22.01	2.32	1.88%	1.33
Hybrid-GBDT with probe	29.16	33.87%	20.85	1.85	1.71%	1.24
Pure-XGBoost	69.43	50.18%	41.81	3.39	2.69%	1.87
Pure-XGBoost with probe	35.27	35.16%	25.14	2.43	2.16%	1.55
Hybrid-XGBoost	42.70	51.50%	28.67	2.47	1.99%	1.41
Hybrid-XGBoost with probe	25.25	35.25%	18.88	1.78	1.67%	1.21

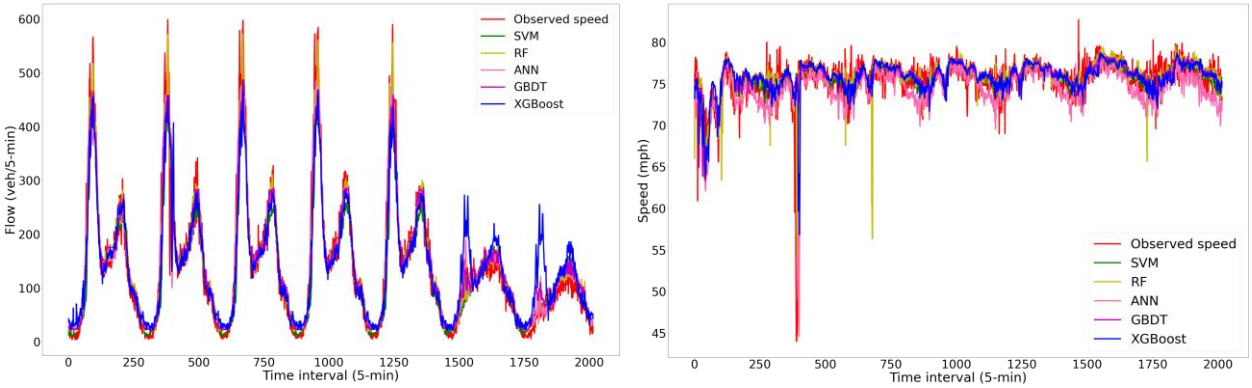
These findings indicate that TSE hybrid-ML models are within the acceptable range (e.g., the flow MAPEs are less than 40%). To further confirm this conclusion, Table 5.4 shows the performance difference between hybrid-ML models and pure-ML models with probe data.

Although pure-ML models with probe data seem to outperform hybrid-ML models in some estimations, performance is quite close to each other. Figure 5.3 shows the comparison of estimated flow and speed by hybrid-ML models to the ground truth, which demonstrates that all hybrid-ML models could accurately estimate speed and flow.

Table 5.4 Performance difference of hybrid-ML vs. pure-ML with probe

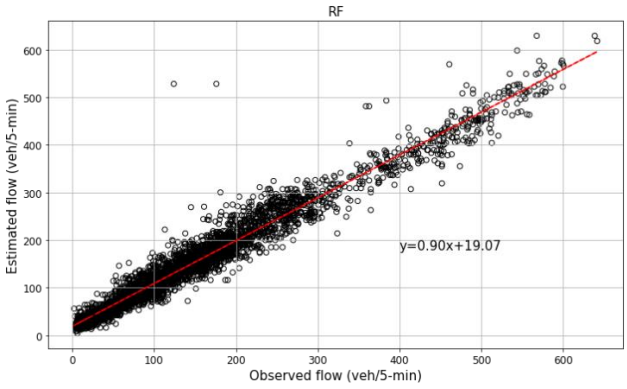
Method	Flow RMSE	Flow MAPE	Flow MAE	Speed RMSE	Speed MAPE	Speed MAE
Hybrid-SVM	16.50%	-22.56	-1.18%	-1.18%	-13.40%	-13.87%
Hybrid-RF	3.50%	1.95%	5.08%	-1.36%	-6.42%	-7.41%
Hybrid-ANN	-1.85%	16.02%	-12.79%	-5.63%	-22.67%	-22.91%
Hybrid-GBDT	9.66%	0.60%	2.41%	-0.43%	-8.51%	-9.77%
Hybrid-XGBoost	17.40	31.73%	12.31%	1.62%	-8.54%	-9.93%

For better presentation, the best flow and speed estimations from the pure-RF with probe data and the hybrid-RF are compared in Figure 5.4. In terms of the resulting trend line, the pure-RF with probe data produces a coefficient of 0.90 and an intercept of 19.07 for flow, and a coefficient of 0.81 and an intercept of 14.69 for speed. The hybrid-RF produces a coefficient of 0.88 and an intercept of 21.12 for flow, and a coefficient of 0.70 and an intercept of 22.48 for speed. All results prove that TSE estimates could be acceptable alternatives when probe data are not available.

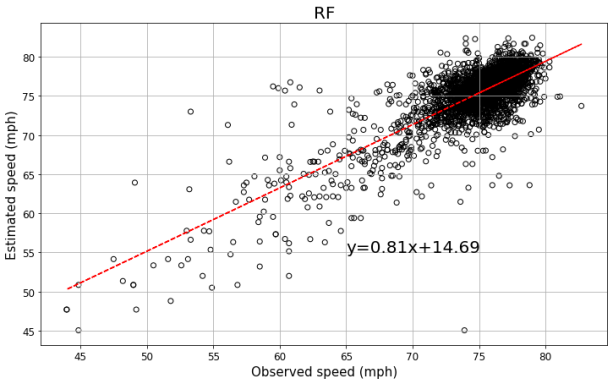


(a) Flow of hybrid-ML (b) Speed of hybrid-ML

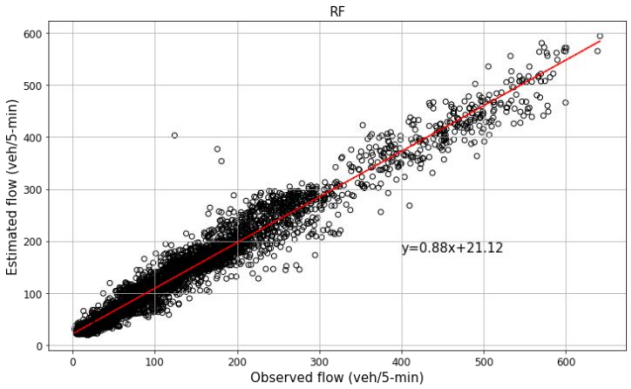
Figure 5.3 Hybrid-ML estimates vs. ground truth



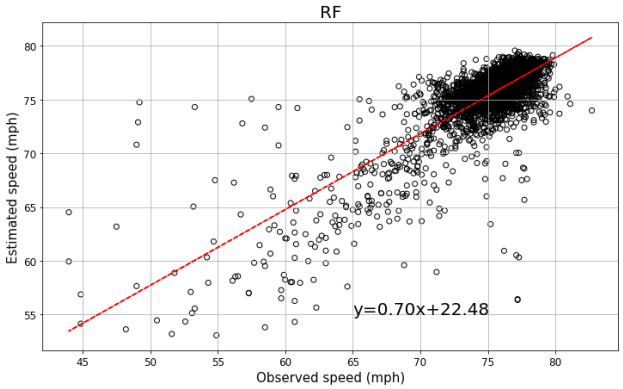
(a) Flow of pure-RF with probe



(b) Speed of pure-RF with probe



(c) Flow of hybrid-RF



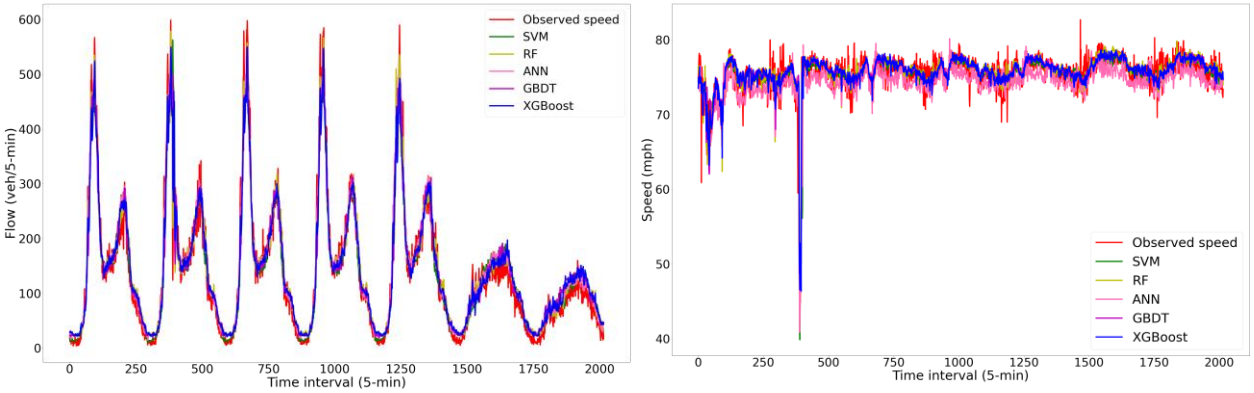
(d) Speed of hybrid-RF

Figure 5.4 Comparison between pure-RF with probe data and hybrid-ML

To further examine whether the TSE accuracy of hybrid-ML can be improved when probe data is also available in training, models of hybrid-ML with probe are also implemented in this research. Table 5.3 also presents performance comparison among hybrid-ML with probe, pure-ML with probe, and hybrid-ML. Table 5.5 provides the percentages of model performance improvement by adding probe data to the training dataset. Figure 5.5 displays the estimated flows and speeds, along with the observed traffic state. From the figure, it can be observed that the models of hybrid-ML with probe can accurately estimate speed and flow patterns. To confirm the observation, the best estimation results from the hybrid-RF with probe are compared with observed data in Figure 5.6. For the obtained trend lines, the coefficient is 0.89 and the intercept is 19.98 for flow and the coefficient is 0.82 and the intercept is 14.23 for speed.

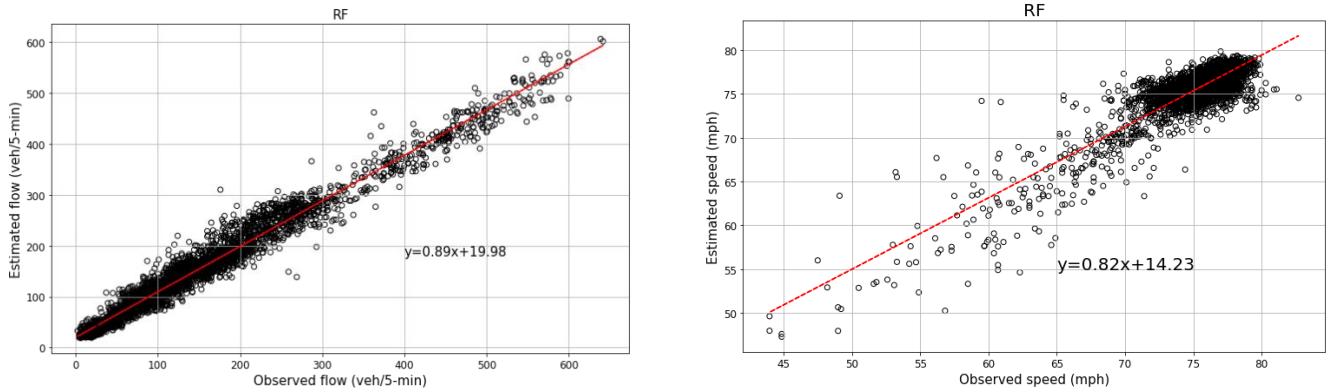
Table 5.5 Percentage of model improvement by adding probe data

Method	Flow RMSE	Flow MAPE	Flow MAE	Speed RMSE	Speed MAPE	Speed MAE
Hybrid-SVM with probe	12.34%	17.25%	18.96%	22.48%	21.82%	20.51%
Hybrid-RF with probe	11.65%	2.88%	5.72%	24.22%	18.59%	18.62%
Hybrid-ANN with probe	22.72%	-3.04%	24.39%	27.67%	26.73%	25.00%
Hybrid-GBDT with probe	1.05%	2.84%	2.93%	20.60%	16.18%	15.07%
Hybrid-XGBoost with probe	28.41%	-0.26%	24.90%	26.75%	22.69%	21.94%



(a) Flow of hybrid-ML with probe (b) Speed of hybrid-ML with probe

Figure 5.5 Hybrid-ML with probe vs. ground truth



(a) Flow of hybrid-RF with probe (b) Speed of hybrid-RF with probe

Figure 5.6 Hybrid-RF with probe vs. ground truth

5.4 Summary

Accurate TSE plays a critical role in the success of ITS on freeways. The accuracy of TSE tends to be affected by the limitation of data quality and quantity. To overcome these issues, this chapter developed a hybrid-ML approach by creating a new training variable based on a second-order macroscopic traffic flow model as a potential replacement of probe data. To evaluate its effectiveness, this chapter conducted a case study on I-215W in Salt Lake County, Utah. The results indicate that the traffic information from prior TFM estimations has good performance and can be used as a supplement or replacement of probe data.

6.0 RECOMMENDATIONS AND IMPLEMENTATION

6.1 Recommendations

As shown in previous chapters, the data from ClearGuide (see Figure 6.1) have low resolution and could be biased. Hence, before implementing such data into daily operational tasks, it is recommended that a more comprehensive data quality study be completed. If data quality is below the expectation, the ML models developed in this research could be a good option to yield more accurate traffic state estimations.

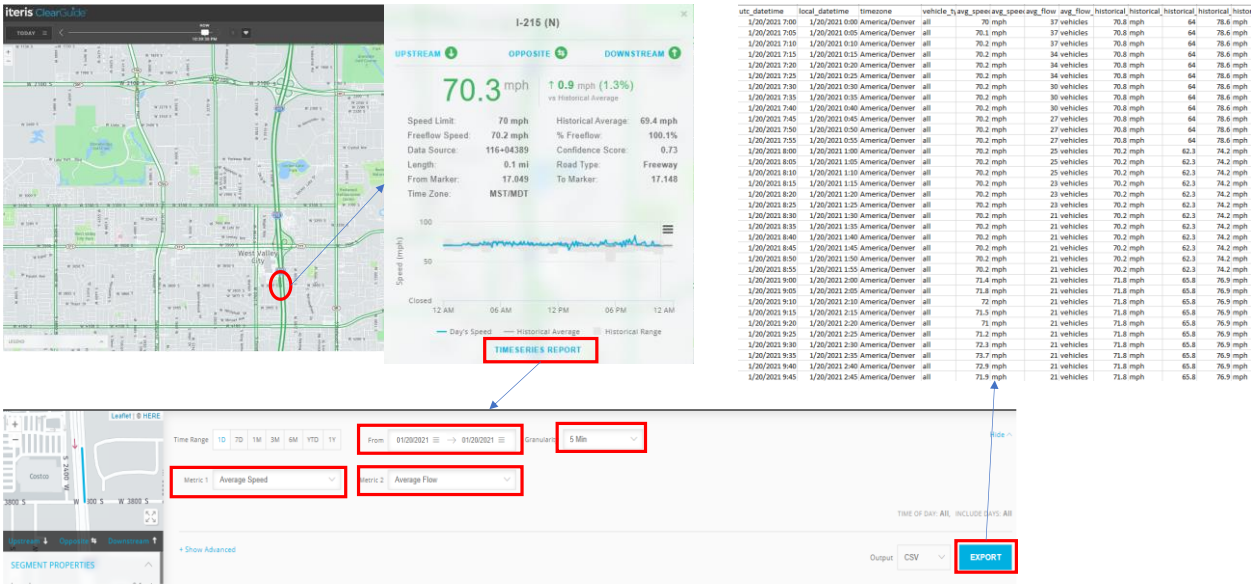


Figure 6.1 Retrieved data from ClearGuide

6.2 Implementation Plan

As all implemented models in this research are quite mature and much open-source information is available online, this chapter develops a work plan for implementing them into the current data visualization and sharing platform used by UDOT's TOC. The implementation plan includes the following seven steps:

1. Retrieve data from both ClearGuide and PeMS databases
2. Create a new database that labels ClearGuide data as "Input" and PeMS data as "output"

3. Use the input and output data to train ML models
4. Implement the trained models to estimate the flow and speed
5. Save the model estimations in the database and label them as “estimates”
6. Fuse the “estimates” with PeMS data to create full-field traffic information for the freeways
7. Visualize the traffic information

A protype of the integrated data visualization platform is shown in Figure 6.2, which includes three tabs: ClearGuide, PeMS, and ML-data.

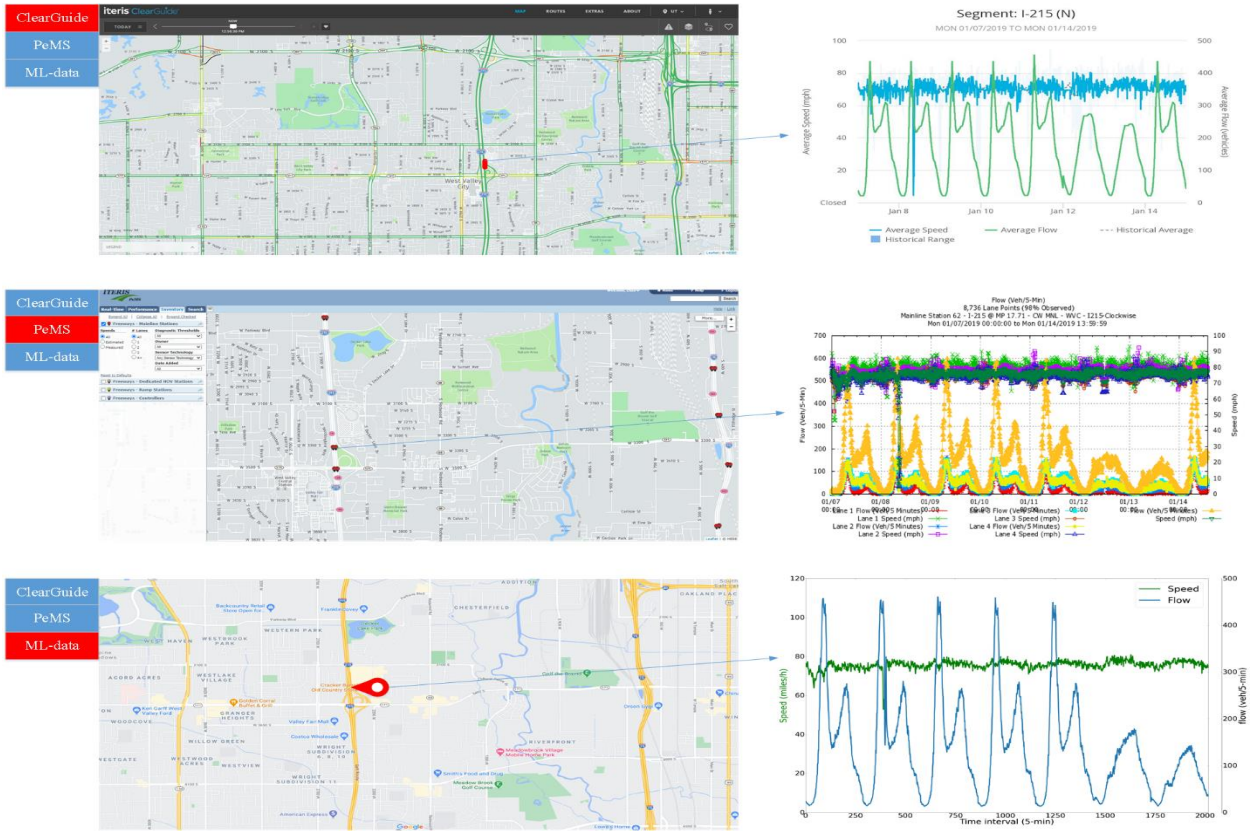


Figure 6.2 ML algorithm integration prototype

7.0 CONCLUSIONS

7.1 Summary

Accurate and statewide traffic information is critical for the success of ITS. Based on the literature review and available data sources, this research first proposed a set of models based on five different ML algorithms to estimate the traffic speed and flow when stationary detector data are not available. Model evaluation results indicated that those ML models fail to produce acceptable traffic state estimates when only taking time and space as input. Further tests with ClearGuide data confirmed the necessity and effectiveness of adding probe data as training input.

However, probe data may not be available in some cases. To overcome this problem and further improve the TSE accuracy, this research also introduced a new hybrid ML framework that integrates the second-order macroscopic traffic flow model. More specifically, the estimates of the traffic flow model were treated as the replacement of the probe data. With a comprehensive numerical test, the model evaluation results show that such hybrid ML models can yield compatible TSE compared with the pure ML that adopts probe data. Hence, when probe data are not obtainable to help with ML training, the proposed hybrid model could be an effective alternative.

Recognizing the limitations of ClearGuide and PeMS data, this research also proposed a work plan that can integrate the developed ML models into the current data visualization and sharing platform owned by UDOT. The model integration would require the coordination of both PeMS and ClearGuide databases as the ML models require data from both for training.

7.2 Contributions

The key contributions of this research are summarized as follows:

- A novel data-driven approach (pure-ML TSE) was developed to obtain accurate and statewide traffic information over the freeway network, using both station-based and GPS-based data. This approach can be viewed as a template for applying regression ML models to TSE.

- A hybrid-ML approach, which introduces the theoretical foundations of traffic flow to ML-based methodologies, was proposed to improve TSE accuracy and overcome the data availability problem. It combines the advantages of both model-driven and data-driven approaches for TSE.

7.3 Limitations and Challenges

The proposed ML models have three general limitations: (1) the developed models can only be implemented at nearby locations and may not be transferable to other freeway segments; (2) the ML models need to be retrained when the original data become dated; and (3) the models can't provide interval-based estimates.

REFERENCES

Allström, A., Ekström, J., Gundlegård, D., Ringdahl, R., Rydergren, C., Bayen, A. M., & Patire, A. D. (2016). Hybrid approach for short-term traffic state and travel time prediction on highways. *Transportation Research Record*, 2554(1), 60-68.

Anusha, S. P., Anand, R. A., & Vanajakshi, L. (2012). Data fusion based hybrid approach for the estimation of urban arterial travel time. *Journal of Applied Mathematics*, 2012.

Asif, M. T., Dauwels, J., Goh, C. Y., Oran, A., Fathi, E., Xu, M., ... & Jaillet, P. (2013). Spatiotemporal patterns in large-scale traffic speed prediction. *IEEE Transactions on Intelligent Transportation Systems*, *15*(2), 794-804.

Aw, A. A. T. M., & Rascle, M. (2000). Resurrection of "second order" models of traffic flow. *SIAM journal on applied mathematics*, *60*(3), 916-938.

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5-32.

Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794).

Daganzo, C. F. (1994). The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, 28(4), 269-287.

Ding, C., Wang, D., Ma, X., & Li, H. (2016). Predicting short-term subway ridership and prioritizing its influential factors using gradient boosting decision trees. *Sustainability*, *8*(11), 1100.

Duan, Y., Lv, Y., Liu, Y. L., & Wang, F. Y. (2016). An efficient realization of deep learning for traffic data imputation. *Transportation research part C: emerging technologies*, *72*, 168-181.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational statistics & data*

analysis, *38*(4), 367-378.

Fu, R., Zhang, Z., & Li, L. (2016, November). Using LSTM and GRU neural network methods for traffic flow prediction. In *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)* (pp. 324-328). IEEE.

Gers, F. A., Schmidhuber, J., & Cummins, F. (1999). Learning to forget: Continual prediction with LSTM.

Göttlich, S., Ziegler, U., & Herty, M. (2013). Numerical discretization of Hamilton--Jacobi equations on networks. *Networks & Heterogeneous Media*, 8(3), 685.

Hamner, B. (2010, December). Predicting travel times with context-dependent random forests by modeling local and aggregate traffic flow. In *2010 IEEE International Conference on Data Mining Workshops* (pp. 1357-1359). IEEE.

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

Hofleitner, A., Herring, R., & Bayen, A. (2012). Arterial travel time forecast with streaming data: A hybrid approach of flow modeling and machine learning. *Transportation Research Part B: Methodological*, 46(9), 1097-1122.

Jia, X., Karpatne, A., Willard, J., Steinbach, M., Read, J., Hanson, P. C., ... & Kumar, V. (2018). Physics guided recurrent neural networks for modeling dynamical systems: Application to monitoring water temperature and quality in lakes. *arXiv preprint arXiv:1810.02880*.

Kang, D., Lv, Y., & Chen, Y. Y. (2017, October). Short-term traffic flow prediction with LSTM recurrent neural network. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)* (pp. 1-6). IEEE.

Karlaftis, M. G., & Vlahogianni, E. I. (2011). Statistical methods versus neural networks in transportation research: Differences, similarities and some insights. *Transportation Research Part C: Emerging Technologies*, *19*(3), 387-399.

Karpatne, A., Watkins, W., Read, J., & Kumar, V. (2017). Physics-guided neural networks (pgnn): An application in lake temperature modeling. *arXiv preprint arXiv:1710.11431*.

Kumar, S. V., Chaitanya Dogiparthi, K., Vanajakshi, L., & Subramanian, S. C. (2017). Integration of exponential smoothing with state space formulation for bus travel time and arrival time prediction. *Transport*, 32(4), 358-367.

Kumar, K., Parida, M., & Katiyar, V. K. (2013). Short term traffic flow prediction for a non urban highway using artificial neural network. *Procedia-Social and Behavioral Sciences*, *104*(2), 755-764.

Leshem, G., & Ritov, Y. (2007, January). Traffic flow prediction using adaboost algorithm with random forests as a weak learner. In *Proceedings of world academy of science, engineering and technology* (Vol. 19, pp. 193-198). Citeseer.

Lederman, R., & Wynter, L. (2011). Real-time traffic estimation using data expansion. *Transportation Research Part B: Methodological*, 45(7), 1062-1079.

Lebacque, J. P. (1996). The Godunov scheme and what it means for first order traffic flow models. In Transportation and traffic theory. *Proceedings of the 13th international symposium on transportation and traffic theory, Lyon, France, 24-26 JULY 1996.*

Lebacque, J. P., Mammar, S., & Salem, H. H. (2007). Generic second order traffic flow modelling. In *Transportation and Traffic Theory 2007. Papers Selected for Presentation at ISTTT17Engineering and Physical Sciences Research Council (Great Britain) Rees Jeffreys Road FundTransport Research FoundationTMS ConsultancyOve Arup and Partners, Hong KongTransportation Planning (International) PTV AG*.

Li, L., Li, Y., & Li, Z. (2013). Efficient missing data imputing for traffic flow by considering temporal and spatial dependence. *Transportation research part C: emerging technologies*, *34*, 108-120.

Lighthill, M. J., & Whitham, G. B. (1955). On kinematic waves II. A theory of traffic flow on long crowded roads. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, *229*(1178), 317-345.

Lingras, P., Sharma, S., & Zhong, M. (2002). Prediction of recreational travel using genetically designed regression and time-delay neural network models. *Transportation Research Record*, *1805*(1), 16-24.

Luo, X., Li, D., Yang, Y., & Zhang, S. (2019). Spatiotemporal traffic flow prediction with KNN and LSTM. *Journal of Advanced Transportation*, *2019*.

Lv, Y., Duan, Y., Kang, W., Li, Z., & Wang, F. Y. (2014). Traffic flow prediction with big data: a deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, *16*(2), 865-873.

Ma, X., Ding, C., Luan, S., Wang, Y., & Wang, Y. (2017). Prioritizing influential factors for freeway incident clearance time prediction using the gradient boosting decision trees method. *IEEE Transactions on Intelligent Transportation Systems*, *18*(9), 2303-2310.

Ma, X., Tao, Z., Wang, Y., Yu, H., & 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.

Mackenzie, J., Roddick, J. F., & Zito, R. (2018). An evaluation of HTM and LSTM for short-term arterial traffic flow prediction. *IEEE Transactions on Intelligent Transportation Systems*, *20*(5), 1847-1857.

Michalopoulos, P. G., Yi, P., & Lyrintzis, A. S. (1993). Continuum modelling of traffic dynamics for congested freeways. *Transportation Research Part B: Methodological*, 27(4), 315-332.

Ni, D., & Leonard, J. D. (2005). Markov chain monte carlo multiple imputation using bayesian networks for incomplete intelligent transportation systems data. *Transportation research record*, *1935*(1), 57-67.

Papageorgiou, M., Blosseville, J. M., & Hadj-Salem, H. (1990). Modelling and real-time control of traffic flow on the southern part of Boulevard Peripherique in Paris: Part I: Modelling. *Transportation Research Part A: General*, *24*(5), 345-359.

Papageorgiou, M., Blosseville, J. M., & Hadj-Salem, H. (1989). Macroscopic modelling of traffic flow on the Boulevard Périphérique in Paris. *Transportation Research Part B: Methodological*, *23*(1), 29-47.

Park, D., Rilett, L. R., & Han, G. (1999). Spectral basis neural networks for real-time travel time forecasting. *Journal of Transportation Engineering*, *125*(6), 515-523.

Payne, H. J. (1971). Models of freeway traffic and control. Mathematical Models of Public Systems. Simulation Councils.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. *the Journal of machine Learning research*, *12*, 2825-2830.

Polson, N., & Sokolov, V. (2017a). Bayesian particle tracking of traffic flows. *IEEE Transactions on Intelligent Transportation Systems*, 19(2), 345-356.

Polson, N. G., & Sokolov, V. O. (2017b). Deep learning for short-term traffic flow prediction. *Transportation Research Part C: Emerging Technologies*, 79, 1-17.

Richards, P. I. (1956). Shock waves on the highway. *Operations research*, *4*(1), 42-51.

Sharmila, R. B., Velaga, N. R., & Kumar, A. (2019). SVM-based hybrid approach for corridor-level travel-time estimation. *IET Intelligent Transport Systems*, 13(9), 1429-1439.

Seo, T., Bayen, A. M., Kusakabe, T., & Asakura, Y. (2017). Traffic state estimation on highway: A comprehensive survey. *Annual reviews in control*, *43*, 128-151.

Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and computing*, *14*(3), 199-222.

Tan, H., Feng, G., Feng, J., Wang, W., Zhang, Y. J., & Li, F. (2013). A tensor-based method for missing traffic data completion. *Transportation Research Part C: Emerging Technologies*, 28, 15-27.

Tan, H., Wu, Y., Cheng, B., Wang, W., & Ran, B. (2014). Robust missing traffic flow

imputation considering nonnegativity and road capacity. *Mathematical Problems in Engineering*, *2014*.

Tak, S., Woo, S., & Yeo, H. (2016). Data-driven imputation method for traffic data in sectional units of road links. *IEEE Transactions on Intelligent Transportation Systems*, *17*(6), 1762-1771.

Tang, J., Zhang, G., Wang, Y., Wang, H., & Liu, F. (2015). A hybrid approach to integrate fuzzy C-means based imputation method with genetic algorithm for missing traffic volume data estimation. *Transportation Research Part C: Emerging Technologies*, *51*, 29-40.

Taylor, C., & Meldrum, D. (1995, July). Freeway traffic data prediction using neural networks. In *Pacific Rim TransTech Conference. 1995 Vehicle Navigation and Information Systems Conference Proceedings. 6th International VNIS. A Ride into the Future* (pp. 225-230). IEEE.

Tian, Y., Zhang, K., Li, J., Lin, X., & Yang, B. (2018a). LSTM-based traffic flow prediction with missing data. *Neurocomputing*, *318*, 297-305.

UDOT Freeway PeMS (2019, March 26). URL https://udot.iteris-pems.com/?fwy=15&dir=S&dnode=Freeway&content=elv&tab=stations&pagenum=4

Utah iPeMS, (2019, March 23). URL https://udot3p.iteris-pems.com/

Van Lint, J. W. C., Hoogendoorn, S. P., & van Zuylen, H. J. (2005). Accurate freeway travel time prediction with state-space neural networks under missing data. *Transportation Research Part C: Emerging Technologies*, *13*(5-6), 347-369.

Van Lint, J. W. C., Hoogendoorn, S. P., & van Zuylen, H. J. (2002). Freeway travel time prediction with state-space neural networks: modeling state-space dynamics with recurrent neural networks. *Transportation Research Record*, *1811*(1), 30-39.

Vapnik, V. (2013). *The nature of statistical learning theory*. Springer science & business media.

Vlahogianni, E. I., Karlaftis, M. G., & Golias, J. C. (2014). Short-term traffic forecasting: Where

we are and where we're going. *Transportation Research Part C: Emerging Technologies*, *43*, 3-19.

Wang, D., Zhang, Q., Wu, S., Li, X., & Wang, R. (2016, August). Traffic flow forecast with urban transport network. In *2016 IEEE International Conference on Intelligent Transportation Engineering (ICITE)* (pp. 139-143). IEEE.

Wang, F. Y. (2010). Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications. *IEEE Transactions on Intelligent Transportation Systems*, *11*(3), 630-638.

Wang, J., Chen, R., & He, Z. (2019). Traffic speed prediction for urban transportation network: A path based deep learning approach. *Transportation Research Part C: Emerging Technologies*, *100*, 372-385.

Wang, J., & Shi, Q. (2013). Short-term traffic speed forecasting hybrid model based on chaos–wavelet analysis-support vector machine theory. *Transportation Research Part C: Emerging Technologies*, *27*, 219-232.

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.

Whitham, G.B. (1975). Linear and nonlinear waves. Modern Book Incorparated.

Wong, G. C. K., & Wong, S. C. (2002). A multi-class traffic flow model–an extension of LWR model with heterogeneous drivers. *Transportation Research Part A: Policy and Practice*, 36(9), 827-841.

Wu, C. H., Ho, J. M., & Lee, D. T. (2004). Travel-time prediction with support vector regression. *IEEE transactions on intelligent transportation systems*, *5*(4), 276-281.

Wu, Y., Tan, H., Qin, L., Ran, B., & Jiang, Z. (2018). A hybrid deep learning based traffic flow prediction method and its understanding. *Transportation Research Part C: Emerging Technologies*, *90*, 166-180.

Xiao, J., Wei, C., & Liu, Y. (2018). Speed estimation of traffic flow using multiple kernel support vector regression. *Physica A: Statistical Mechanics and its Applications*, *509*, 989-997.

Yang, B., Sun, S., Li, J., Lin, X., & Tian, Y. (2019). Traffic flow prediction using LSTM with feature enhancement. *Neurocomputing*, *332*, 320-327.

Yang, S., Wu, J., Du, Y., He, Y., & Chen, X. (2017). Ensemble learning for short-term traffic prediction based on gradient boosting machine. *Journal of Sensors*, *2017*.

Yin, W., Murray-Tuite, P., & Rakha, H. (2012). Imputing erroneous data of single-station loop detectors for nonincident conditions: Comparison between temporal and spatial methods. *Journal of Intelligent Transportation Systems*, 16(3), 159-176.

Yu, B., Yang, Z. Z., Chen, K., & Yu, B. (2010). Hybrid model for prediction of bus arrival times at next station. *Journal of Advanced Transportation*, 44(3), 193-204.

You, J., & Kim, T. J. (2000). Development and evaluation of a hybrid travel time forecasting model. *Transportation Research Part C: Emerging Technologies*, 8(1-6), 231-256.)

Yuan, Y., Yang, X. T., Zhang, Z., & Zhe, S. (2020). Macroscopic Traffic Flow Modeling with Physics Regularized Gaussian Process: A New Insight into Machine Learning Applications. *arXiv preprint arXiv:2002.02374*.

Zeng, D., Xu, J., Gu, J., Liu, L., & Xu, G. (2008, August). Short term traffic flow prediction using hybrid ARIMA and ANN models. In *2008 Workshop on Power Electronics and Intelligent Transportation System* (pp. 621-625). IEEE.

Zhang, H. M. (2002). A non-equilibrium traffic model devoid of gas-like behavior. *Transportation Research Part B: Methodological*, *36*(3), 275-290.

Zhang, J., Wang, F. Y., Wang, K., Lin, W. H., Xu, X., & Chen, C. (2011). Data-driven intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 12(4), 1624-1639.

Zhang, K., Zheng, L., Liu, Z., & Jia, N. (2020). A deep learning based multitask model for network-wide traffic speed prediction. *Neurocomputing*, *396*, 438-450.

Zhang, Y., & Ge, H. (2013). Freeway travel time prediction using Takagi–Sugeno–Kang fuzzy neural network. *Computer-Aided Civil and Infrastructure Engineering*, *28*(8), 594-603.

Zhang, Y., & Haghani, A. (2015). A gradient boosting method to improve travel time prediction. *Transportation Research Part C: Emerging Technologies*, *58*, 308-324.

Zhang, Y., & Liu, Y. (2009). Traffic forecasting using least squares support vector machines. *Transportmetrica*, *5*(3), 193-213.

Zhang, Z., & Yang, X. (2020). Freeway traffic speed estimation by regression machine learning techniques using probe vehicle and sensor detector data. *Journal of transportation engineering, Part A: Systems*, *146*(12), 04020138.

Zhang, Z., Yuan, Y., & Yang, X. (2020). A Hybrid Machine Learning Approach for Freeway Traffic Speed Estimation. *Transportation Research Record*, *2674*(10), 68-78.

Zhong, M., Lingras, P., & Sharma, S. (2004). Estimation of missing traffic counts using factor, genetic, neural, and regression techniques. *Transportation Research Part C: Emerging Technologies*, *12*(2), 139-166.

Zou, Y., Zhu, X., Zhang, Y., & Zeng, X. (2014). A space–time diurnal method for short-term freeway travel time prediction. *Transportation Research Part C: Emerging Technologies*, *43*, 33-49.

Zhu, L., Guo, F., Polak, J. W., & Krishnan, R. (2018). Urban link travel time estimation using traffic states-based data fusion. *IET Intelligent Transport Systems*, 12(7), 651-663.

Zhang, Z., & Yang, X. (2020). Freeway traffic speed estimation by regression machine learning techniques using probe vehicle and sensor detector data. *Journal of transportation engineering, Part A: Systems*, 146(12), 04020138.