TRAVEL TIME FORECASTING ON A FREEWAY CORRIDOR: A DYNAMIC INFORMATION FUSION MODEL BASED ON MACHINE LEARNING APPROACHES

by

Bo Qiu

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Infrastructure and Environmental Systems

Charlotte

2021

Approved by:

____

Dr. Wei Fan

____

Dr. Martin Kane

____

Dr. David Weggel

____

Dr. Jay Wu

____

Dr. Jiancheng Jiang

ABSTRACT

BO QIU. Travel Time Forecasting on a Freeway Corridor: a Dynamic Information Fusion Model based on Machine Learning Approaches. (Under the direction of DR. WEI FAN)

Metropolitan areas suffer from frequent road traffic congestion not only during peak hours but also during off-peak periods. Currently, the increasing availability of vehicle probe data has made the real-time travel time prediction a reality. The traffic on freeways is complicated to interpret, which can be impacted by various traffic features, many of which are also unpredictable. Despite the difficulties, a more profound understanding of the change of travel times and the TTP will greatly help infrastructure design, traffic management and operations, and transportation related decision-makings.

Various statistical methods and machine learning methods have been employed in travel time forecasting. However, such machine learning methods practically face the problem of overfitting. Tree-based ensembles have been applied in various prediction fields, and such approaches usually produce high prediction accuracy by aggregating and averaging individual decision trees. The inherent advantages of these approaches can not only help obtain better prediction results but also have an excellent bias-variance tradeoff, which can help avoid overfitting. To improve the accuracy and the interpretability of the model, the random forest (RF) method is developed and used to analyze and model the travel time on freeways in this research. However, when the travel time prediction (TTP) time horizon increases (i.e., greater than 15 min), the performance of the RF method decreases significantly. Recently, as another powerful prediction method, the Long Short-Term Memory (LSTM) neural network methods have been widely applied to short-term traffic prediction. In this research, the attention mechanism (AM) is implemented by developing the neural network to capture the inner relationship within the traffic data. The proposed LSTM with attention mechanism (LSTM_AM) method achieves its superior capability for TTPs longer than 15 minutes (i.e., from 30 min to 60 min), overcoming the performance issue through long temporal dependency and memory blocks. To validate the accuracy and reliability of proposed models, the proposed approaches are tested using a freeway corridor in Charlotte, North Carolina, using the probe vehicle-based traffic data. The input features are introduced in detail, and data preprocessing is also presented. The mean absolute percentage errors (MAPEs) are computed for different observation segments in varying prediction horizons ranging from 15 to 60 minutes to measure the effectiveness of the proposed TTP algorithms. The features' relative importance values show that variables (such as travel time 15 minutes before and time of day) have the highest contribution to the predicted results. The results also indicate that the proposed TTP models perform better in prediction at the 15-minute interval than the other time horizons. Besides, the RF model has the best prediction performance with an average MAPE of 6.34% in the 15-minute prediction horizon, and the LSTM_AM model has the best performance in all other prediction horizons (including 30 min, 45 min, and 60 min). In practice, they can be applied in their preferred prediction horizons. A comparison with other prediction methods validates that the proposed RF and LSTM methods can achieve a better prediction performance in both accuracy and efficiency, suggesting that they can be used as a part of the successful solutions to address critical and real-world transportation challenges.

ACKNOWLEDGEMENTS

First, I would like to express my most profound appreciation to my advisor Professor Wei Fan for his continuous support of my Ph.D. study and related research. I am grateful for his patience, motivation, and immense knowledge. His guidance helped me in researching and writing this dissertation.

I would also like to thank the rest of my thesis committee: Professor Martin Kane, Professor David Weggel, Professor Jay Wu, and Professor Jiancheng Jiang, for their insightful comments and questions on this dissertation, which inspired me to widen my research from various perspectives.

I am grateful to my groupmates in the CAMMSE lab: Dr. Zhen Chen, Dr. Pengfei Liu, Dr. Yang Li, Dr. Zijing Lin, Mr. Shaojie Liu, and Mr. Li Song for their help in my life and study. The days and nights we spent discussing and addressing problems together gave me immense support, which I will keep in mind.

Last but not least, I would like to thank my family: my wife Shan Hu, my two sons, and my daughter for supporting me spiritually throughout writing this thesis and my life in general.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AM	Attention Mechanism
ANN	Artificial Neural Networks
ARIMA	Autoregressive Integrated Moving Average
CAMMSE	Center for Advanced Multimodal Mobility Solutions and Education
DOW	Day of Week
FOSIM	Freeway Operations Simulation
K-NN	K-Nearest Neighbor
LR	Linear Regression
LSTM	Long Short-Term Memory
MAPE	Mean Absolute Percentage Error
NN	Neural Network
RF	Random Forest
RIV	Relative Important Value
RMSE	Root Mean Square Error
SVM	Support Vector Machine
SVR	Support Vector Regression
TMS	Traffic Management Systems
TOD	Time of Day
TTP	Travel Time Prediction
XGB	eXtreme Gradient Boosting

CHAPTER 1: INTRODUCTION

1.1. Problem Statement

In 2006, the average one-way travel time for the commuters in the U.S. was 25.0 minutes. The U.S. census bureau showed that the average one-way commute in the United States increased to a new high of 27.6 minutes in 2019. The increase of about 2.6 minutes from 2006 to 2019 represents an increase of about 10% over 13 years. The increase in commuting time is due to urban expansion to some extent, but the increase of traffic congestion time per capita proves that congestion contributes significantly and forms the main component. Furthermore, congestion not only increases travel times but also decreases travel time reliability. This uncertainty entails additional costs to travelers and hence to society. Therefore, there is a growing demand for TTP and its high accuracy. However, TTP is influenced by a wide range of traffic, weather, and other features. Metropolitan areas are suffering frequent road traffic congestion in both peak hours and off-peak periods. Accurate and reliable TTP in freeway networks is a critical component that will be helpful to all modes of transportation in all urban, suburban, and rural areas. It is widely accepted that considerable accuracy and reliability of TTP are highly desired by both travelers and transportation planners. Therefore, the capability to forecast dynamically changing traffic conditions, particularly travel times, is of utmost importance in a wide range of traffic management applications to relieve its negative impact on society, environment, and economy. Accurate TTP can significantly help enhance the performance of the traffic management systems (TMS), in which travelers are given the opportunities to react to the traffic proactively (Oh et al., 2015).

A prediction refers to a calculation or an estimation that uses data from previous events, combined with recent trends to come up with a future event outcome. TTP is a challenging problem because of the traffic and events' underlying hidden patterns. Furthermore, the required data is not always available. Traditional data sources for travel time are gathered from cameras and traffic sensors, and the high costs of installation and maintenance restrict the usage of this equipment to only the major roads.

Another challenge is to select a suitable and efficient prediction method. Most existing machine learning methods can capture the nonlinear pattern of travel time but suffer from over-fitting problems. The acquisition and popularization of big data in transportation have enabled the collection and diffusion of real-time traffic information. Different researchers have employed various machine learning approaches, and the prediction accuracy indicates that machine learning algorithms have better performances than traditional statistical models. However, such machine learning methods are practically faced with an overfitting problem that is difficult to overcome, and especially when the traffic conditions significantly change, the prediction results are often unsatisfactory. In addition, the RF method has a perfect Bias-Variance tradeoff which can help avoid the overfitting problem. This research proposes RF TTP prediction models to predict the short-term travel time on freeways by using the probe vehicle-based traffic data and therefore helps to gain a better understanding of how different contributing factors might affect travel time on freeways. As another powerful method, neural network (NN) methods have been extensively applied to short-term traffic prediction in the past few years. However, when the traffic conditions change considerably, the prediction results are often unsatisfactory. In this research, the second machine learning method LSTM_AM method is developed to capture nonlinear properties within the traffic data. Compared to traditional NN, the LSTM_AM can overcome the performance issue through memory blocks and achieve its superior capability for time series prediction with long temporal dependency. To validate the effectiveness of the proposed TTP methods, the proposed approaches are tested using a freeway corridor in Charlotte, North Carolina, using the probe vehicle-based traffic data. A comparison with the proposed methods and baseline methods is made. The results indicate that the proposed methods achieve a better prediction performance in both the accuracy and stability.

1.2. Research Objectives

The research objectives of this study are listed as follows:

1. To develop the TTP methods using appropriate and cutting edge machine learning-based approaches to improve the prediction accuracy;
2. To conduct a case study to evaluate and test proposed prediction models and compare their performance in different prediction time horizons; and
3. To systematically analyze significant impact factors and investigate their relative importance.

1.3. Expected Contributions

This study develops TTP methods on freeways based on the RF and LSTM machine learning methods. The expected contribution of this study can be summarized as follows:

1. Develop different machine learning-based TTP methods; and
2. Validate and compare the proposed TTP models.

1.4. Research Overview

The research is structured as follows. In this chapter, the introduction and motivation of the TTP, the research objectives, and expected contribution have been discussed.

Chapter 2 presents a literature review on the state-of-the-art and state-of-the-practice of the short-term TTP. Machine learning (including data-based parametric models and non-parametric models) based traffic prediction methods will be described.

Chapter 3 introduces the Regional Integrated Transportation Information System (RITIS) data set that is used to analyze the TTP. The RITIS traffic data is introduced, and detailed information about raw weather data is discussed. Temporal and spatial traffic data and historical weather data are combined with the RITIS traffic data. The traffic and weather data combination process is also presented and discussed in this chapter.

Chapter 4 presents the TTP methodologies, including RF and LSTM. Two deep machine learning-based TTP models (e.g., RF and LSTM) are developed based on the RITIS and weather sample dataset. The model development process is also presented, including the data structure, parameter determination, model training, and model validation.

Chapter 5 validates the proposed TTP models. For the machine learning prediction model, the data training step is described to determine the parameters in the model structure. Potential factors include but are not limited to the following: TOD, DOW, month, segment characteristics, and weather conditions. The optimization process of the proposed model’s parameters (e.g., the number of maximum features, and the number of hidden layers) is also discussed.

Chapter 6 presents the comparison and evaluation of the proposed TTP models. The MAPE is set as the statistical criterion that is used to measure the prediction error.

Chapter 7 concludes the study with discussions of the developed prediction models, solution approaches, and research results. Suggestions for future research are also provided.

CHAPTER 2: LITERATURE REVIEW

2.1. Introduction

In chapter 2, a review of various aspects of TTP studies, including travel time definitions, TTP methodologies, TTP modeling, analysis, is conducted. Existing and cutting-edge modeling methods for TTP are also introduced.

The following sections in this chapter are presented as follows. Section 2.2 discusses the travel time definitions and TTP classification methods. Section 2.3 presents the review of current TTP methods, including statistical methods and machine learning methods. Section 2.4 concludes this chapter with a summary.

2.2. Background

2.2.1. Travel Time Definition

Travel time is defined as the total time for a vehicle to travel from one point to another over a specified route (Zhu et al., 2009).

2.2.2. Classification Approach

TTPs can be categorized from different perspectives, and the most popular classification method is to classify them according to their prediction horizon as short, medium, and long-terms (Oh et al., 2018). Van Lint (2004) defined the short-term TTP as a 0-60 minute interval. It was found that identifying a appropriate time horizon in TTP plays the most significant role in the TTP applications (Shen, 2008). The second critical perspective is the road network category, including either arterial roads or freeways. Researchers considered the flows, speeds, densities, and travel time in short-term traffic flow prediction as an essential component of the intelligence traffic system application (Liu et al., 2017). It was more complicated to predict the travel time on urban signalized arterial roads due to the presence of signals and intersections (Oh et al., 2018).

2.3. Travel Time Prediction Approach

Thanks to the integration of big data and transportation management, different approaches have been developed and applied in this area. The techniques can be divided into two general groups: statistical methods and machine learning methods. On the statistical side, linear regression (LR) and time series have been widely applied in TTP as in other research areas. Furthermore, among the time series model, the autoregressive integrated moving average (ARIMA) model has been frequently deployed in TTP based on historical traffic data. Machine learning methods are considered more effective, accurate, and feasible. Different machine approaches (such as support vector, tree-based ensemble learning, and recurrent neural network) have been applied in TTP areas by various researchers. The prediction results indicated that compared with the traditional statistical models, machine learning methods have a significant improvement not only in prediction accuracy but also in time efficiency (Mori et al., 2015).

In the last several decades, research on more reliable short-term travel time forecasting has attracted numerous researchers, from transportation engineers to data scientists. The machine learning (data-based) traffic prediction methods can be divided into two major categories: parametric models and non-parametric models (Van, 2004). Parametric models are always model-based methods, where all of the parameters can be estimated with empirical data, and the model structure is predetermined based on certain theoretical assumptions. LR is the most typical parametric model, where the dependent variable is a linear function of the explanatory (independent) input variables. The input variables are typically traffic observations in several past time intervals. Bayesian net is the second type of parametric model, where the explanatory variables are assumed to be conditionally independent, given the target variables. The third type of parametric model is the time series model, a series of data points indexed in time order. Time series forecasting involves the use of a model to predict future values based on previously observed values. In the last two decades of the last century, the ARIMA model has been the most widely used one for TTP. The first application of ARIMA in traffic analysis dates back to 1979 (Ahmed & Cook, 1979). For parameter-based approaches, real-time data integrity is also a critical factor in determining the prediction accuracy since many model-based systems deal with feeding data in real-time for online services.

In the non-parametric models, both the structure of the model and the parameters are not predetermined. However, the term “non-parametric” does not mean that there are no parameters in the models to be estimated. Furthermore, the number and typology of the parameters are unknown a priori and sometimes uncountable. Due to the rapid development of data science, non-parametric estimation methodologies are being quickly updated. One of the most popular ones in the literature of TTP is the artificial neural networks (ANN). Due to their ability to capture complex relationships in large data sets, ANN methods have been widely used in travel time forecasting (Dharia & Adeli, 2003). As the typical non-parametric models, ANN can be developed without being given a specific form of the function. Furthermore, the restrictions on the multicollinearity of the explanatory variables can be partially overcome. Different types of neural networks have been applied to TTP, such as the regular multilayer feedforward neural networks (Yildirimoglu & Geroliminis, 2013) and spectral basis neural networks (Park & Rilett, 1999). The input variables selection is different, which depends on the data availability and the model training process. Different variations of the backward algorithm can carry out different types of neural networks. Support vector machine (SVM) methods are another choice for TTP. This advanced algorithm consists of the decision function, the kernel functions’ application, and the sparsity of solutions. The SVM model is suitable for TTP based on historical travel time data. Some researchers

(Yildirimoglu & Geroliminis, 2013; Wu et al, 2004) used SVM methods to estimate travel time. In the calculation process, the algorithm maps the input data into a higher dimensional space by the kernel function. The process stops after finding the flattest linear function related to the transferred input vectors (i.e., when the target variable has an error smaller than a predefined threshold). This linear function can be mapped again into the initial space and get the final nonlinear function which is used for TTP. Both the ANN and SVM models tend to be overfitting due to the complicated structure and the many parameters that need to be calibrated, which is a serious problem commonly existing in the non-parameter machine learning algorithm.

In the TTP applications, another popular non-parametric approach is the local regression approach, which can yield accurate prediction results. In the local regression approach, the algorithm chooses a set of historical data which have similar characteristics to the current situation. The prediction results base themselves on generating a model constructed by the selected data set. The local regression models' types depend on the techniques used to select the set of similar historical points and the methodology chosen to fit the model.

Some semi-parametric models have been developed in traffic time prediction, which is a combination of parametric and non-parametric methods. The main idea of the semi-parametric method is to loosen some of the assumptions of the parametric model to obtain a more flexible structure (Ruppert et al., 2003). In the case of applications, semi-parametric models are presented in the form of varying coefficient regression models. Travel time can be calculated by a linear function of the naive historical and instantaneous predictors. Furthermore, the parameters differ depending on the departure time interval and prediction horizon (Schmitt & Jula, 2007).

With the wide applications of machine learning algorithms in TTP, different approaches have been deployed in different areas with varying data sources. The methodologies that researchers have used include, but are not limited to, the following: neural network approach (e.g., SSNN, LSTM), nearest neighbor approach (e.g., k nearest neighbor), SVM, and ensemble learning approach (e.g., RF and gradient boosting). Some research efforts are listed as follows, and Table 2.1 summarizes the studies reviewed in chronological order.

2.3.1. Nearest Neighbors Method

2.3.1.1. Myung et al.'s research work

In 2011, Myung et al. deployed the k nearest neighbor (KNN) approach to predict travel time using the data provided by two data sources (i.e., the automatic toll collection system and the vehicle detector system). By combining these two sets of data, the limitations of each dataset can be minimized by the model, and the prediction's accuracy can be enhanced. The authors also compared the proposed KNN TTP method with other TTP models by using the above datasets. The comparison results proved the KNN feasibility for TTP.

2.3.1.2. Yu et al.'s research work

In 2017, Yu et al. applied a hybrid prediction method (RF and KNN) to bus TTP. The proposed hybrid method was compared with other TTP methods, which included linear regression, KNN, SVM, and RF. The prediction results showed that the proposed hybrid TTP methods yield the best prediction accuracy and efficiency performance.

2.3.1.3. Moonam et al.'s research work

In 2019, Moonam et al. applied multiple machine learning methods, which included KNN, least squares regression boosting, and Kalman filter (KF) in TTP. The comparison consisted of the link and corridor of the freeway and concluded that the KF algorithm reached the best prediction performance (i.e., smallest MAPEs).

2.3.2. Support Vector Regression Method

2.3.2.1. Wu et al.'s research work

In 2004, Wu et al. first deployed the Support Vector Regression (SVR) model in TTP and compared the results with time series methods (as a baseline) using real freeway traffic data. For a given training dataset, SVR had a significant ability to locate the global minima. In the model comparison, the SVR had better prediction accuracy than the time series approach. The prediction results also indicated that the SVR predictor significantly reduces prediction errors. This study revealed that the SVR method is suitable for TTP and can improve prediction performance.

2.3.3. Ensemble Learning Method

2.3.3.1. Hamner et al.'s research work

In 2011, Hamner et al. researched to improve the RF TTP accuracy with the data collected from the GPS simulation dataset. The authors developed a context-dependent RF method. In the validation, the proposed model obtained a reasonable prediction performance, in which the result showed the root mean square error (RMSE) of the model was smaller than 7.5%.

2.3.3.2. Zhang and Haghani's research work

In 2015, Zhang and Haghani (2015) proposed an advanced tree-based ensemble learning method, and the gradient boosting regression tree was applied to forecast the travel time on freeways. The experiment dataset was collected from the freeway in Maryland, U.S. Different combinations were used to conduct sensitivity analyses and test the effect of the parameters in the gradient boosting regression tree. The prediction results showed that the performance of the proposed model is considerably good in the freeway TTP application.

2.3.3.3. Li and Bai's research work

In 2016, Li and Bai collected trajectory data of freight vehicles in Ningbo, China. The authors applied a gradient boosting regression tree in TTP. Bayesian optimization was employed for model fitting in this research. The conducted experiment results indicated that the proposed model is suitable for real-world applications.

2.3.3.4. Gupta et al.'s research work

In 2018, Gupta et al. applied RF and gradient boosting models in taxi TTP in Porto. The taxi trajectory data was collected and used for TTP. The model comparison results showed that the gradient boosting model performance is superior to the RF model.

2.3.3.5. Chen and Fan's research work

In 2019, Chen and Fan employed the advanced ensemble learning method XGB model to forecast travel time variability and freeway travel time in Charlotte, NC, U.S. The data was collected from RITIS and combined with weather data. The prediction results showed that the XGB performs very well in the TTP application.

2.3.4. Neural Network Method

2.3.4.1. Park and Rilett's study

In 1999, Park and Rilett developed a BP NN to forecast travel time on freeway links. The freeway link travel time was collected in Houston, Texas, and the dataset collected from the automatic vehicle identification (AVI) system was used as the validation database. The proposed BP neural network achieved a reasonable prediction accuracy, in which the MAPEs ranged from 7.4% to 18%.

2.3.4.2. Van Lint et al.'s research work

In 2002, Van Lint et al. researched to apply the state space neural network (SSNN) in TTP. The data was collected from freeway operations simulation (FOSIM). In the validation and model training process, 80% of data was used to train the model, and 20% of data was used to validate the model. In the model training process, the authors also ranked each variable's relative importance value (RIV) and kept the most significant parameters to improve the model efficiency.

2.3.4.3. Wisitpongphan et al.'s study

In Wisitpongphan et al.'s (2012) study, the authors applied a BP NN model in TTP for the freeway link. The data was collected from the selected label probes vehicles via GPS service in Thailand, and the 297 vehicles' one-month trajectories were collected as the sample data. The prediction results proved that the proposed model is suitable for predicting travel time based on the GPS dataset, which achieved an average mean squared error (MSE) of less than 3%.

2.3.4.4. Zheng and Van Zuylen's study

In 2013, Zheng and Van proposed a TTP model based on the multiple layer neural network, and the authors conducted experiments by collecting the traffic data along with probe data. In the validation process, the authors used the data simulated from VISSIM simulation software. The prediction results proved that the proposed Artificial Neural Network (ANN) model achieves a good prediction performance and is suitable in real-world practice.

2.3.4.5. Duan et al.'s study

In 2016, Duan et al.'s research group first applied LSTM neural network model in the TTP area. The authors conducted the experiment with the data collected from the freeways in England. The authors optimized the steps ahead of the TTP and found 1-step ahead of TTP can achieve the optimization prediction results.

2.3.4.6. Liu et al.'s study

In 2017, Liu et al. continued experimenting with LSTM application in TTP. The authors developed different hyper-parameters in TTP based on the data collected from the freeways in

California, U.S. The proposed model was compared with the ARIMA model and other models, and the comparison results indicated that the LSTM has superior performance in prediction accuracy.

2.3.4.7. Wang et al.'s study

In 2018, Wang et al. collected the database from floating-car. The authors applied multiple different state-of-the-art machine learning methods to conduct the TTP. The database from the floating-car included more than one million historical traffic data, and the results indicated the proposed three layer neural network achieves the best prediction accuracy.

Table 2.1 Summary of TTP approaches

Year	Author	Country/City	Roadway Category	Data Source	Method Category	Data Type	Prediction method
2000	Wunderlich et al.	N/A	N/A	Simulated data (INTEGRATION)	Native model	Travel time	Exponential filtering
2002	Dion et al.	Virginia, US	N/A	Simulated data (INTEGRATION)	Traffic theory-based model	Travel time	Delay models
2002	Van Lint et al.	N/A	Freeway	Simulated data (FOSIM)	Non-parametric	Travel time, travel speed	State-Space Neural Network
2005	Wu et al.	Taiwan	Freeway	Loop detector	Non-parametric	Travel time	SVR
2007	Schmitt and Jula	California, US	Urban road	Loop detector	Native model	Travel time	Switch model
2008	Zou et al.	Maryland, US	Freeway	Roadside detector	Hybrid non-parametric	Travel time	Combined Clustering Neural Networks
2009	Li et al.	Atlanta, US	N/A	Simulated data (VISSIM)	Hybrid non-parametric	Travel time	Combined Boosting and Neural Network
2010	Papageorgiou et al.	N/A	N/A	Simulated data (MATANET)	Traffic theory-base model	Travel time	Macroscopic Simulation
2010	Hamner et al.	N/A	Freeway	GPS	Non-parametric	Travel speed	RF
2011	Myung et al.	Korea	Freeway	ATC system	Non-parametric	Travel time	KNN
2012	Wisitpongphan	Bangkok, Thailand	Freeway	GPS	Non-parametric	Travel time	BP Neural Network
2013	Yildirimoglu & Geroliminis	California, US	Freeway	Loop detector	Hybrid non-parametric	Travel time	Combined Gaussian Mixture, PCA, and Clustering
2015	Zhang and Haghani	Maryland, US	Interstate freeway	INRIX	Non-parametric	Travel time	Gradient boosting
2015	Joao et al.	Porto, Portugal	Urban road	STCP system	Hybrid non-parametric	Travel time	Combined RF, Projection Pursuit Regression, and SVM
2016	Duan et al.	England	Freeway	Cameras, GPS, and loop detectors	Non-parametric	Travel time	LSTM Neural Network
2016	Li and Bai	Ningbo, China	Urban	Trajectory data	Non-parametric	Truck trajectory, travel time, travel speed	Gradient boosting
2017	Liu et al.	California, US	Interstate freeway	PeMS	Non-parametric	Travel time	LSTM Neural Network

Year	Author	Country/City	Roadway Category	Data Source	Method Category	Data Type	Prediction method
2017	Fan et al.	Taiwan	Freeway	Electric toll	Non-parametric	Travel time, vehicle information	RF method
2018	Wang et al.	Beijing, China	Urban road	Floating-Car Data	Non-parametric	Taxi trajectory data	LSTM
2018	Wei et al.	China	Urban road	Vehicle passage records	Non-parametric	Travel time	LSTM
2018	Wang et al.	Beijing and Chengdu, China	Urban road	GPS	Non-parametric	Travel time	LSTM
2018	Gupta et al.	Porto, Portugal	Urban road	GPS	Non-parametric	Taxi traveltime	RF and gradient boosting
2019	Moonam et al.	Madison, Wisconsin, US	Freeway	Bluetooth detector	Non-parametric	Travel time	KNN, KF
2019	Kumar et al.	Chennai, India	Urban road	GPS	Non-parametric	Travel time	KNN
2019	Cristobal et al.	Gran Canaria, Spain	Urban road	Public transport network	Non-parametric	Travel time	K-Medoid Clustering Technique
2019	Ran et al.	England	Freeway	Freeway Record	Non-parametric	Travel time	LSTM
2020	Kwak & Gerolimin is	California, US	Freeway	PeMS	Parametric	Travel time	Dynamic linear model
2020	Fu et al.	Beijing, Suzhou, China	Urban road	Ride-hailing platform	Non-parametric	Travel time	Graph attention network
2021	Chiabaut & Faitout	Lyon, French	Freeway	Loop detector	Non-parametric	Travel time	PCA and Clustering

2.4. Summary

A literature review and synthesis of the existing research related to traditional statistical TTP and machine learning-based TTP methodologies have been presented and introduced in the preceding sections.

CHAPTER 3: DATA COLLECTION AND PREPARATION

3.1. Introduction

In this chapter, the data collection and data processing will be discussed, and the raw travel time data gathered from the RITIS will be combined with historical weather data to generate the dataset needed in this research. Section 3.2 introduces the raw travel time data source collected from RITIS. In section 3.3, the weather data collection and categorization are presented, and section 3.4 discusses the data combination and preparation process. In the end, section 3.5 summarizes the chapter with a short conclusion.

3.2. Data Collection

The travel time dataset is collected from the RITIS in this research. RITIS is an advanced traffic system that includes segment analysis, probe data analytics, and signal analytics. I-485 is one of the busiest interstate freeways in Charlotte, which loops encircling the city. A series of segments in the southern loop are selected for the case study. To achieve an acceptable accuracy of prediction, the model has to be well-established with large historical data that need to be secured which typically contain at least one year's data (Torday, 2010). In this research, the dataset is collected from 01/01/2019-12/31/2019, and the time interval is 15 minutes, which has uninterrupted coverage in the RITIS data with 24 hours per day (365 days a year). The selected study section starts from the interchange with I-77 (Exit 67) and ends at the interaction with US-74 (Exit 51). Figure 3.1 shows the study road segments and three traffic message record sensors (A, B, C) that are selected for the model validation. There are 37 miles of roadways and 32 traffic message channel code segments in the clockwise and counterclockwise directions.

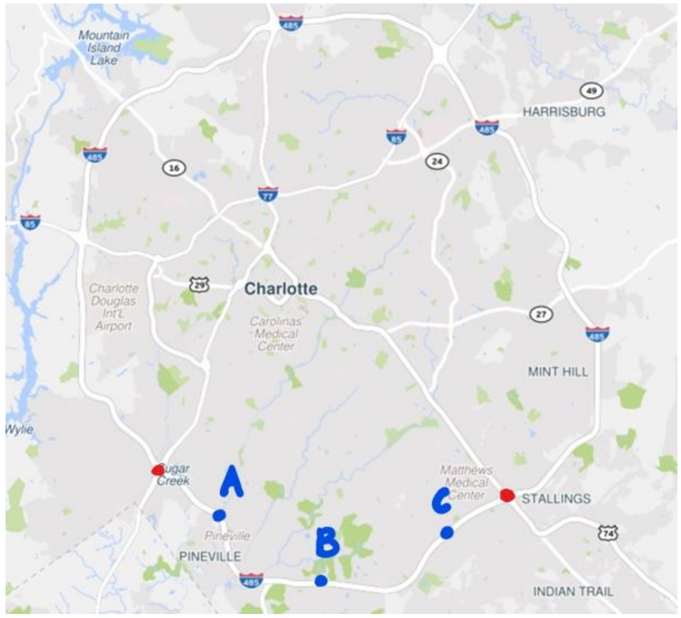

Figure 3.1 Selected road segments on southern I-485

In this research, the raw weather data are collected at locations close to the Charlotte Douglas International airport, which is not far from the selected roadway segments. The basic weather data include weather indexes (including the temperature, dew point, humidity, air pressure, visibility, wind speed, wind direction, gust speed, precipitation, and weather conditions). Table 3.1 shows a sample of raw weather data.

Table 3.1 Weather data sample

Date	Time	Conditions
Saturday, Oct 5th, 2019	7:55 AM	Rain
Saturday, Oct 5th, 2019	8:55 AM	Rain
Saturday, Oct 5th, 2019	9:55 AM	Light Rain
Saturday, Oct 5th, 2019	10:55 AM	Light Rain
Saturday, Oct 5th, 2019	11:55 AM	Light Rain
Saturday, Oct 5th, 2019	12:55 AM	Light Rain
Saturday, Oct 5th, 2019	13:55 PM	Light Rain

It was found that the travel time reliability is sensitive to the weather condition and severe weather (Zhao & Chien, 2012). The weather can significantly affect the travel time and speed, which are two crucial traffic flow parameters of transportation, resulting in the deterioration of a traffic system's performance (Koetse & Rietveld, 2007). Since the weather data were recorded on a per hour basis, the discrepancy in the time intervals is treated by developing and using a mapping method to combine the travel time data with the weather conditions. The original weather conditions from raw weather data are initially classified into 30 detailed weather conditions. To improve the computing power of the model, the weather conditions are further classified into three general groups (i.e., normal, rain, and snow/fog/ice) in this research (Chen and Fan, 2019). Table 3.2 provides the classification of the newly grouped weather conditions.

Table 3.2 Newly grouped weather conditions

Snow/fog/ice	Normal	Rain
Haze	Light Rain	Clear
Fog	Rain	Partly Cloudy
Smoke	Heavy Rain	Mostly Cloudy
Patches of Fog	Light Drizzle	Scattered Clouds
Mist	Heavy Thunderstorm	Overcast
Shallow Fog		Unknown
Light Freezing R	Light Thunderstorm	
Light Ice Pellet	Thunderstorm	
Light Freezing D	Drizzle Squalls	
Light Freezing F		
Ice Pellets		
Light Snow		
Snow		
Heavy Snow		

3.3. Data Structure and Preprocessing Steps

There are so many factors that can affect travel time, including the traffic volume, speed, road class, and occupancy, event, accident information, segment locations, and weather conditions.

In this research, the raw sample dataset is collected from the southern part of the I-485 freeway, which is divided into 32 sections by the recorded sensor segment. The raw travel time data include 4 main attributes and the TMC-code indicates the segment ID, timestamp, speed, travel time in seconds.

According to the travel time pattern and literature review, the data structure is designed as shown in this Figure 3.2. Current travel time is the travel time at the prediction segment 15 minutes before the target prediction horizon, which is called T_{t-1}; The lag1_travel time is the travel time at prediction segment 30 minutes before, which is called T_{t-2}.

$T_{t-\mathrm{w}}$ (the travel time at prediction segment one week before) is the first introduced to the TTP-related features. However, the result shows that it is a significant feature and has high related importance value in the modeling process. *Time dif 1* is the travel time change value at $\mathrm{T}_{\mathrm{t}-1}$, and the calculation process is shown in Figure 3.2. The upper segment travel time is the travel time of the nearest upstream road segment 15 minutes before, which is called T_{t-1}^{i-1}; The upper segment travel time *dif 1* is the travel time change value at the nearest upstream road segment 15 minutes before.

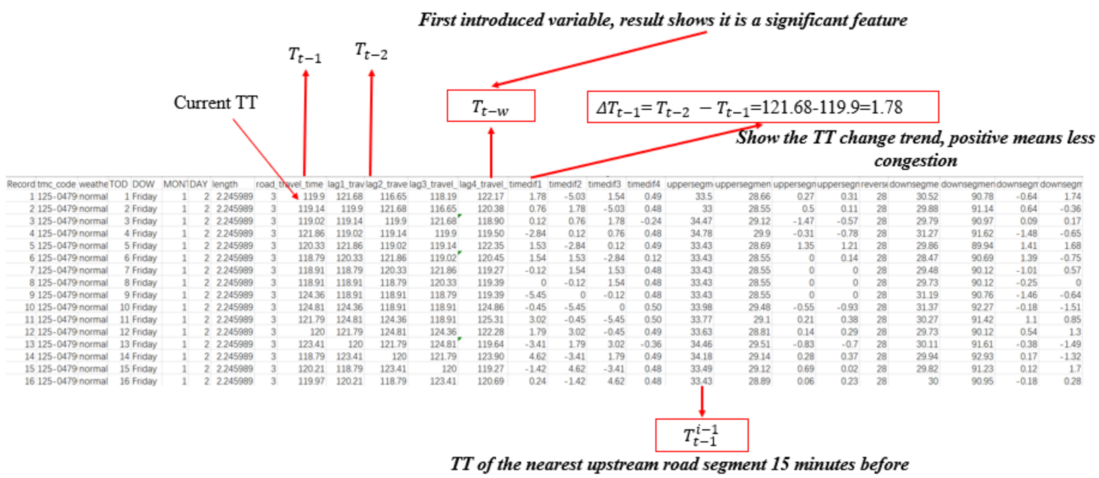

Figure 3.2 Travel time dataset structure

The other traffic information on each segment (from the sensor to sensor) contains the subject segment and adjacent segment travel times, day of week (DOW), time of day (TOD), segment length, and space mean speed. The missing data rate of the sample dataset in this research is less than 0.5% (i.e., 4246 out of 981,083), and the missing values are replaced with the mean value of its two closest values. A well-defined preprocessing capability that corrects various types of data errors, including missing data, is mandatory for a reliable TTP system with acceptable accuracy and efficiency (Oh et al., 2018). In this research, the proposed prediction model is developed under normal traffic conditions and does not consider the factor of unexpected conditions (e.g., serious accidents or special events). While implementing the proposed model is feature selection, as the sample dataset contains more than 980,083 observations and 35 features. It is well known that some features are not very relevant. However, no one has the domain expertise to confirm which feature is applicable. In the machine learning application, the predictor features (variables) usually significantly affect the prediction results. Testing the effect of the individual variable is essential to understand the performance of each feature, in which higher relative importance presents a more substantial impact in TTP. Table 3.3 indicates the definitions and attributes of selected features. Figure 3.2 shows the data pre-preparation process.

Table 3.3 Definitions of selected variables

Variable	**Attribute**	**Definition**
ID	Categorical	Road segment ID
T_t	Float	The travel time at the prediction road segment
Speed	Float	Space Mean Speed
TOD	Categorical	Time of day (indexed 1 to 96, represent the time from 0:00-24:00 by every 15-minute timestep)
DOW	Categorical	Day of the week (indexed 1 to 7, represent from Monday through Sunday)
Month	Categorical	Month (indexed 1 to 12, represent from January to December)
Weather	Categorical	Weather (indexed 1 to 3, represent normal, rain and snow/ice/fog)
T_{t-1}	Float	The travel time at the prediction segment 15 minutes before
T_{t-2}	Float	The travel time at the prediction segment 30 minutes before
T_{t-w}	Float	The travel time at the prediction segment one week before
ΔT_{t-1}	Float	The ravel time change value at T_{t-1}
ΔT_{t-2}	Float	The ravel time change value at T_{t-2}
ΔT_{t-w}	Float	The travel time change value at T_{t-w}
T_{t-1}^{i-1}	Float	The travel time of the nearest upstream road segment 15 minutes before
T_{t-1}^{i-2}	Float	The travel time of the second nearest upstream road segment 15 minutes before
ΔT_{t-1}^{i-1}	Float	The travel time change value at the nearest upstream road segment 15 minutes before
ΔT_{t-1}^{i-2}	Float	The travel time change value at the second nearest upstream road segment 15 minutes before
T_{t-1}^{i+1}	Float	The travel time of the nearest downstream road segment 15 minutes before
T_{t-1}^{i+2}	Float	The travel time of the second nearest downstream road segment 15 minutes before

ΔT_{t-1}^{i+1}	Float	The travel time change value at the nearest downstream road segment 15 minutes before
ΔT_{t-1}^{i+2}	Float	The travel time change value at the second nearest downstream road segment 15 minutes before

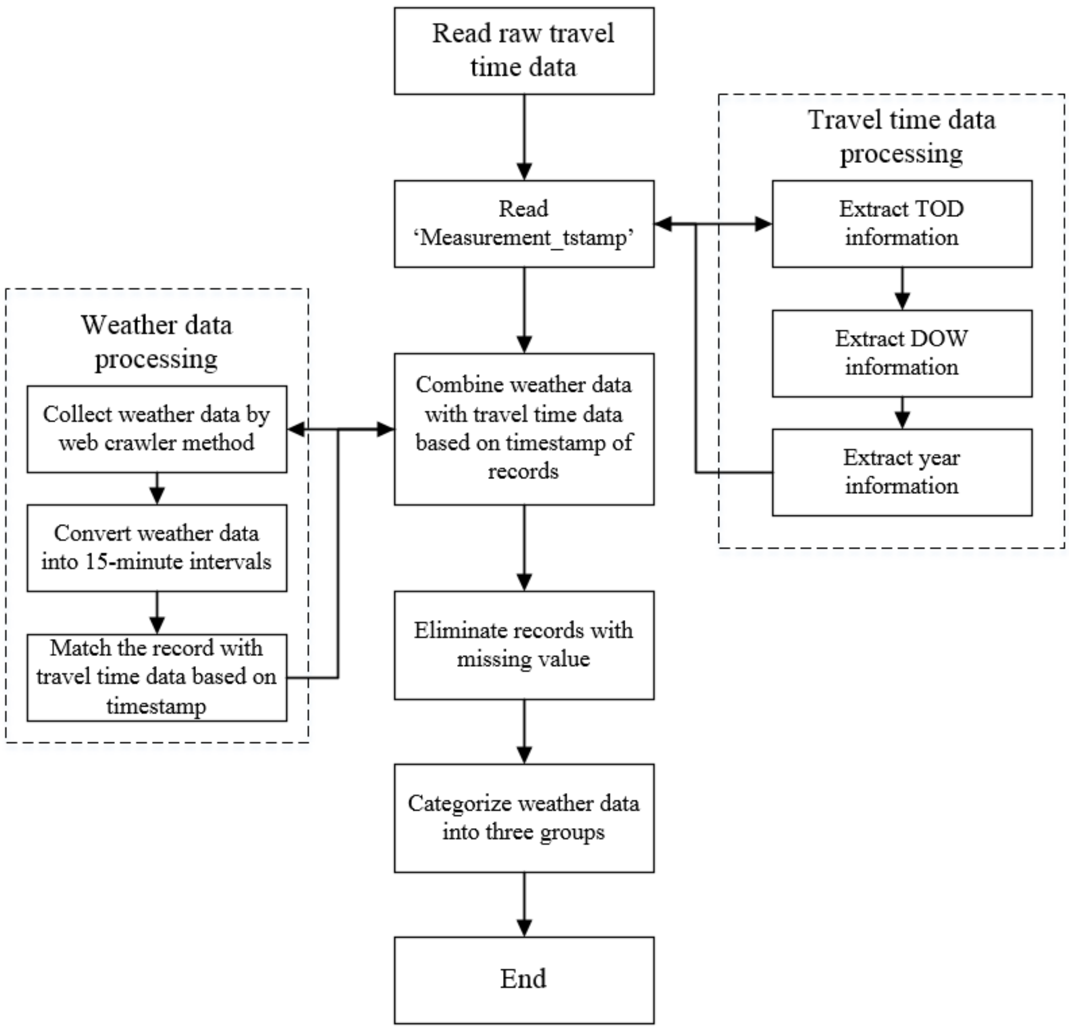

Figure 3.3 Data preparation steps

3.4 Summary

In chapter 3, the data gathering and collection method is introduced, and the data structure and pre-preparation approach to combine the travel time with original weather data is also discussed. This preparation provides a clean database for developing TTP in future tasks.

CHAPTER 4: TTP METHODOLOGY

Chapter 4 discusses the TTP methodology based on the dataset collected and presented in Chapter 3. Two machine learning-based TTP models (RF and LSTM_AM) are developed. The detailed characteristics of each model will be described, including the data structure configuration and parameter determination.

4.1. Random Forest Algorithm

4.1.1. Ensemble Learning Methodology

An ensemble is a supervised learning algorithm because it can be trained and then used to make predictions. The ensemble learning methods consist of multiple single tree-based models (e.g., decision tree model), each alternatively solving the problem. The prediction results tend to be more accurate when substantial diversity among the models exists (Kuncheva and Whitaker, 2003). Decision trees (DT), RF, and Boosting are among the top sixteen data science and machine learning tools used by data scientists. The three methods are similar, with a significant amount of overlapping. DT always suffers from high variance, making the estimation results fragile to the specific training data used. Building multiple models from samples of the original training data, called bagging, can reduce this variance. However, the bagging can make the trees highly correlated. RF is an extension of bagging in addition to building trees based on multiple samples of the original training data. It also constrains the features that can be used to build the trees, forcing trees to be different. The RF models have been widely applied to various research fields (Greenhalgh and Mirmehdi, 2012; Xu et al., 2016). For classification tasks, RF typically gives high accuracy while also having a faster classification time. An RF classifier requires training with large datasets, which in our study are available due to the nature of the travel record data collected.

Furthermore, the RF computational process runs efficiently on large data sets, reducing model complexity, overcoming overfitting, and improving efficiency. As known, the overfitting comes from the estimated model that fits the training data too well. In other words, overfitting is caused by the model function being complex to consider each data point and even outliers. The RF algorithm can build a large number of random trees and then combine the results from each individual tree. The benefit of using the RF methods is that through averaging, the variance can be reduced.

4.1.2. Random Forest Algorithm

RF is an algorithm that can compete with gradient enhancement trees in integrated learning, especially for its convenient parallel training, which is very tempting in the era of big data and large samples. For each tree, the feature selection is conducted randomly. The prediction process is shown in Figure 4.1. The difference between the RF algorithm and the decision tree algorithm is that in RF, finding the root node and splitting the feature nodes will run randomly.

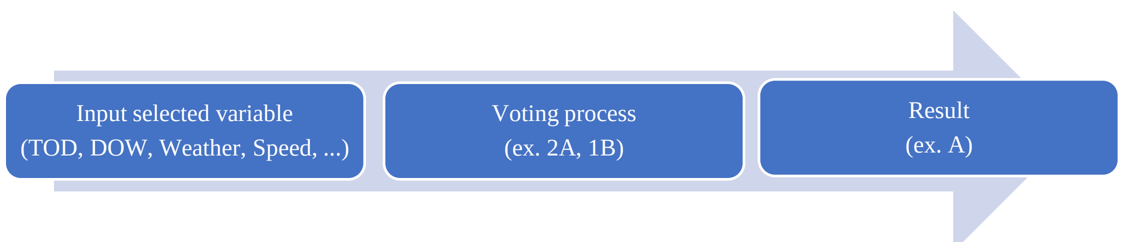

Figure 4.1 Prediction process of the RF algorithm

Figure 4.2 shows the prediction process of the RF algorithm, which is described as follows.

(1) The number of training data points is N, and the number of variables in the classifier is M.

(2) Select the m variables in the whole variable set M to determine the decision at a node of the tree. (Note that m is always considerably smaller than M)

(3) To construct the forest by trees, choose a training set *k* times with replacement from all *N* training datasets. Each of these datasets is called a bootstrap dataset. The number *k* is the number of the trees to be trained.

(4) For each tree node, randomly choose *m* variables on which to make the decision at that node. Calculate and get the best split based on these *m* variables in the training set.

(5) The "*Gini Index*" is used to calculate the Gini value to determine the best split point, which can describe the purity after the split. The *Gini index* will fall between 0 and 1, and the smaller the value, the better the split. If a dataset contains elements from two classes, the *Gini index* is defined as follows:

$$Gini(T) = 1 - \sum_{j=1}^{n} (p_j^2) \tag{1}$$

where p^j is the relative proportion of class *j* in the original dataset *T*, and *n* is the number of classes in dataset *T*.

$$Gini_{split}(T) = \frac{N_1}{N} Gini(T_1) + \frac{N_2}{N} Gini(T_2) \tag{2}$$

The randomness in the RF method means two things: n training samples are randomly retracted from the training set and the m feature subsets are randomly drawn from M features. They are called bootstrapping and random feature selection. The two randomnesses are very important to the performance of RF. Bootstrapping ensures that the same data for every tree are not used, which helps the trained model to be less sensitive to the original training data. The random feature selection ensures one to reduce the correlation between trees. If the model is trained by a single DT, the change in one or two data will result in a significant change in the prediction result.

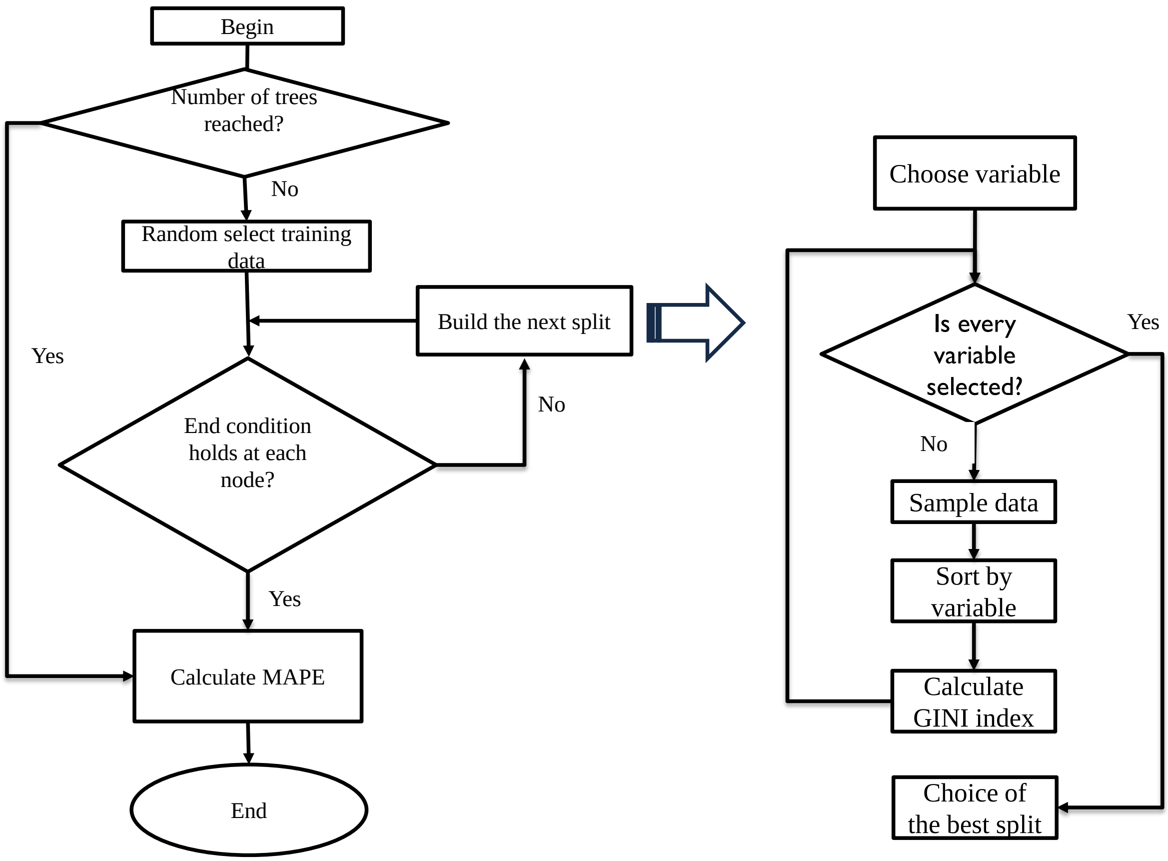

Figure 4.2 RF algorithm processing flow

4.1.3. Extreme Gradient Boosting Method

The extreme gradient boosting (XGB) method is a scalable machine learning method for tree boosting (Chen & Guestrin, 2016). XGB model is an efficient ensemble tree-based algorithm, which has won many machine learning competitions.

As an ensemble tree boosting method, the objective function generates a new classification after each iteration. The classification can be constantly improved as the predictions are made from weak classifiers over the error of the previous classifier. Incorrect classification receives a bigger weight which forces the classifier to focus on their performance in the following iterations. The process leads the classification to develop tree structures perfectly and efficiently. The objective function can be defined as follows. The former part of the function controls the model prediction accuracy, while the latter part balances the complexity. This function represents the training loss and the regularization. The current machine learning algorithm lacks a robust regularization factor, making the learning process overfitting. XGB overcomes this weakness by providing a strong regularization penalty which constrains overfitting (Dong et al., 2018).

The objective function of the XGB model can be presented as below (Chen and Guestrin, 2016):

$$Obj(\Theta) = L(\Theta) + \Omega(\Theta)$$

where,

$L(\Theta)$ = The training loss, which can measure how well the model fits the training dataset;

$\Omega(\Theta)$ = The regularization term, which can measure the complexity of the model.

The loss of the training data can be calculated as:

$$L = \sum_{i=1}^{n} l(y_i, \hat{y}_i)$$

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} \omega^2$$

where,

i = the index of examples;

T = the number of leaves in the tree;

γ = the penalty coefficient to the number of leaves;

λ = the penalty coefficient of regularization; and

ω = the score of leaf j.

For each independent tree structure, T is the number of leaves in the tree. ω is the leaf weight of the tree structure. γ and λ are regularized coefficients.

4.1.4. Proposed RF Approaches

4.1.4.1. Feature Selection and Preprocessing Steps

In the prediction model, the southern part of the I-485 freeway is divided into 32 sections by the recorded sensor segment in this research. Traffic data on each segment (from sensor to sensor) contains information on the subject segment and adjacent segment travel times, DOW, TOD, segment length, and space mean speed. The RITIS real-world travel time data used for this study have a less than 0.5% missing rate (i.e., 4246 out of 981,083). From the previous studies (e.g., Wang et al., 2018), the variables that significantly impact the TTP included the basic variables (such as TOD, DOW, month, and weather) and the spatial and temporal characteristics of the adjacent road segments.

Furthermore, in this research, the travel times (collected several steps ahead of the travel time to be predicted) are also accounted for in the model estimation. The prediction model is developed under normal traffic conditions and does not consider unexpected situations (e.g., special events). The data on each segment will be used to train one forest, which consists of decision trees. The RF model prediction includes two major steps: training and prediction. The forests are constructed using randomly selected parameter combinations and different numbers of trees during the training step.

To achieve the best modeling results, it is vital to test the impact of different combinations of parameters on the RF model prediction performance. Based on previous studies, three features can be tuned to optimize the model's predictive power: Max_features, N_estimators (number of trees), and Min_sample-leaf. They are presented as follows:

Max_features:

This is the maximum number of features in the RF model allowed to try in each individual tree. There are multiple options available in Python to assign maximum features. “Auto/None” is a command that simply takes all the features that make sense in every tree, which simply does not put any restrictions on the individual tree. The “SQRT” option takes the square root of the total number of features in each individual run. For example, if the total number of variables is 100, under this option, the system can only take 10 of them in each individual tree. The “log2” option is another similar type of option used for *max_features*. In this research, after several tests, the random subspace method is applied. The number of features considered at each internal node of random forests is m, which is randomly chosen to be $m = INT(log_2 M + 1)$, where M is the total number of features, as suggested by Breiman (2001a, b).

n_estimators:

This is the number of trees that the model developer wants to build before taking the maximum voting or averages of predictions. A larger number of trees will give one better performance with a compromise of computing efficiency. One should choose a value as high as the processor can handle because this makes the predictions more robust and stable.

min_sample_leaf:

The leaf is the end node of a decision tree. A smaller leaf makes the model more prone to capture noise in the train data. In this research, after several trials of different leaf sizes, a minimum leaf size of 20 is chosen. In addition, researchers have to face the problem named “tuning RF parameters in practice” and the excellent answer to it varies from dataset to dataset. In this research, the tool RandomSearch is applied to optimize the tuning process. One needs to define the range of parameters and then run these procedures to get the best model. In this research, the first run is

1000 trees, with 1/2 features per node. RF models are not sensitive if the features are independent or dependent, though many will perform better if the data are preprocessed. A simple way to identify dependence among features is to calculate a correlation coefficient between each feature and all other features. To determine the importance of the features, one can build a forest and see which features get used, as RF models tend to split out the results by using the most statistically significant features.

It is also important to note that the performance measure used in this research is the mean absolute percentage error (MAPE). The MAPE statistic usually expresses accuracy as a percentage that is calculated as follows:

$$MAPE = \frac{1}{m}\sum_{i=1}^{m}\left|y_i - y_i^{\wedge}\right|$$

where, m = The total number of the data points,

$y_i^{\wedge}$ = The predicted travel time value in the test dataset of record *i*,

y_i = The actual travel time value in the test dataset of record *i*.

4.2. LSTM Algorithm

LSTM is an algorithm that was initially introduced by Hochreiter and Schmidhuber (Koetse & Rietveld, 2007). Different from the standard feedforward Recurrent Neural Network (RNN), LSTM has feedback connections. It can deal with not only single data points but also the sequences of the data. A standard LSTM unit comprises a cell, an input gate, an output gate, and a forget gate. Cell is responsible for remembering values over arbitrary time intervals, while the input, output, and forget gate control the information flow into and out of the cell. LSTM networks are well-suited to classify, process, and make predictions based on time series data since there can be lags of unknown duration between important events in a time series. LSTMs can deal with the vanishing gradient problem, which is hard for modeling using RNN. LSTM cell is different from the recurrent unit, which is a specially redesigned cell memory unit. The cell vectors can encapsulate the information which is assigned to forget part from previously-stored memory, and to add part of the new information. Moreover, as a data-driven approach, LSTM is significantly influenced by historical data since the method is highly dependent on the scale and integrity of the historical data.

4.2.1. Recurrent Neural Network

Neural networks are a set of algorithms, modeled loosely after the human. The human brain is composed of neurons, such as the eyes and the sense of touch. When these neurons receive external stimuli, they will generate electrical signals, which are transmitted layer by layer and output a result through the brain. RNN is a Neural Network used to process sequential data. Compared with ordinary neural networks, it can process sequential data. For example, the meaning of a word may have different meanings depending on the content mentioned above. As the travel time data is highly relevant sequential data, the predicted result has a high relative with a series of time relevant and spatial relevant data. It is more efficient to predict within a context. Event before and after has a high impact on the target prediction interval.

RNN can solve such problems well. It is well understood that the way people understand an article and their understanding of each word will depend on what they have seen before, rather than discarding everything they have seen before, forgetting, and then understanding the word. In other words, there is always continuity in people’s thinking. The inability of traditional neural networks to maintain such continuity is a huge drawback. For example, when people watch a movie, they try to categorize what is happening in each frame. There is no clear way to use the traditional neural network to add events that occurred earlier in the movie to help understand what happened later. However, RNN can do it. As one can see from Figure 4.3, there is a loop of operations that allows them to retain what they have learned.

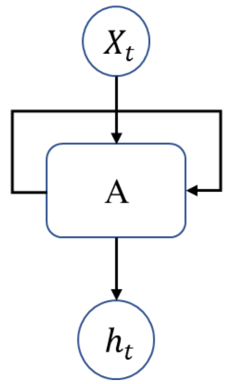

Figure 4.3 RNNs Network

where

X_t: is the input vector;

h_t: is the model output vector.

It is crucial to explore and solve such problems as "long-term dependencies" by choosing the proper parameters to achieve the best modeling results. Unfortunately, in practice, RNNs cannot solve the TTP problem. Bengio et al. (Hochreiter and Schmidhube, 1997) studied this problem and found that RNNs were indeed difficult to solve this problem. Another fatal drawback of RNN is that the model is hard to train as the back propagation neural network optimization process, which is called gradient disappearance and gradient explosion.

4.2.2. Structure of the Memory Unit of LSTM

LSTM is a particular type of RNN, which performs better in longer sequences than traditional RNN. LSTMs have been widely used to solve various problems, and excellent results have been achieved. Specifically, the design of LSTMs aims mainly to avoid the mentioned long-term dependency problem. Their essence is to remember information over a long period, and it can be done very quickly. LSTM treats the hidden layer as a memory unit. Therefore, LSTM has the advantage of dealing with the correlation in both the short and long term. Moreover, the LSTM can automatically determine the optimal time lags. In this research, the structure of the memory unit is shown in Figure 4.4, where the memory cell is at the center of the unit. The input data is X_t and the output is the prediction result Y_t . There are three gates in the memory unit: input gate, forget gate, and output gate (represented by green circles). In this figure, the state of the cell is indicated by S_t, and the input of each of the three gates is the input data X_t. The previous state of the memory cell is defined as S_{t-1}. LSTM cell consists of input layer, hidden layer, output layer and context layer. The input layer receives traffic data (such as speeds and time-related data); The output layer consists of one neuron that calculates the predicted travel time. The context layer stores the previous internal states of the model. In other words, it can store and extract traffic context information. The hidden layer receives inputs from the input layer and then stores them in context layer, finally transforming them into output layer.

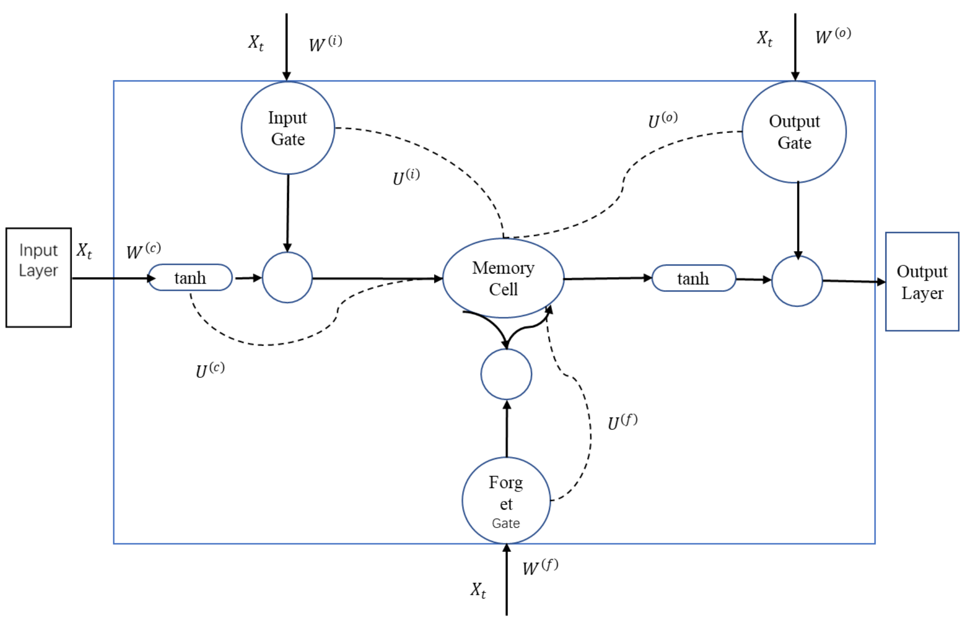

Figure 4.4 Memory unit of LSTM

The mathematic model of LSTM in Figure 4.5 can be indicated by

$$Z_i = \sigma\left(W^{(i)}X_t + U^{(i)}S_{t-1}\right) \tag{3.}$$

$$Z_f = \sigma\left(W^{(f)}X_t + U^{(f)}S_{t-1}\right) \tag{4.}$$

$$Z_o = \sigma\left(W^{(o)}X_t + U^{(o)}S_{t-1}\right) \tag{5.}$$

$$Z = \tanh\left(W^{(c)}X_t + U^{(c)}S_{t-1}\right) \tag{6.}$$

$$S_t = Z_f \odot S_{t-1} + Z_i \odot Z \tag{7.}$$

$$h_t = Z_o \odot \tanh(S_t) \tag{8.}$$

$$y_t = \sigma\left(W^{(h)}h_t\right) \tag{9.}$$

where,

Z_i, Z_f, Z_o: the output of input gate, forget gate, and output gate; After multiplying the weight matrix by the splicing vector, the sigmoid activation function converts it into a value between 0 and 1, which is used as a gated state.

Z: is a value between -1 and 1 through the activation function *tanh*.

S_t : the new state of the memory cell.

h_t: the hidden state of the memory cell.

$W^{(i)}, W^{(f)}, W^{(o)}, W^{(c)}, U^{(i)}, U^{(f)}, U^{(o)}, U^{(c)}$: coefficient matrix.

$\odot$: Hadamard Product.

The complex algorithm of LSTM can be understood in combination with the formulas (1-9) in a simplified way, as shown in Figure 4.5. Compared to RNN, LSTM can improve the prediction accuracy by capturing the correlation within time series in both the short and long term by the function of different gates. The critical point to the LSTMs is the cell state. The LSTM can remove or add information to the cell state, which is regulated by structures called gates.

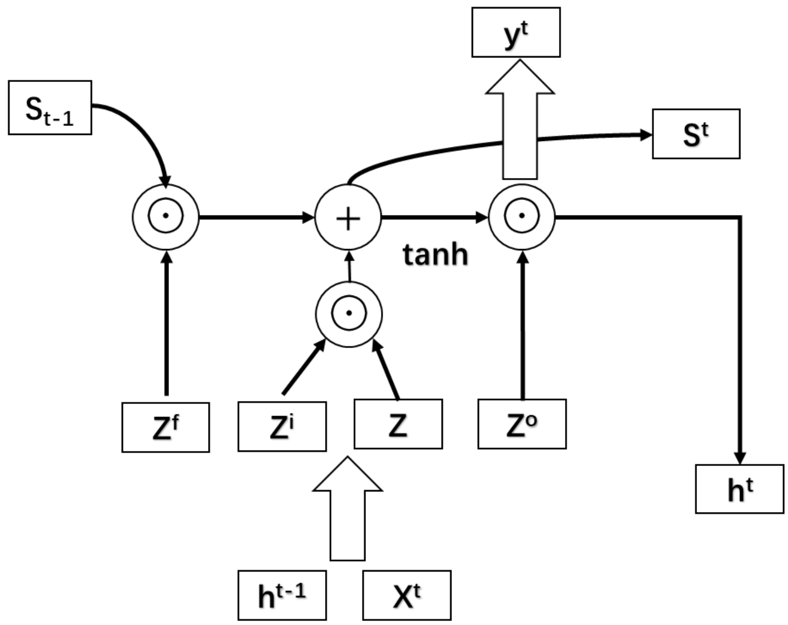

Figure 4.5 The schematic diagram of LSTM

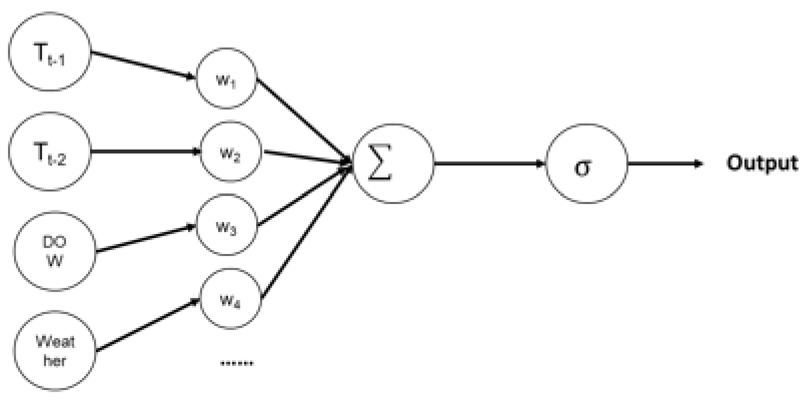

Figure 4.6 The travel time prediction process in one node in neural networks

In Figure 4.6, the diagram shows what one node may look like in TTP in the LSTM cell. A neuron can receive multiple signals, but some are important and some are not so important. Each traffic information as input has a weight that is generated by the sigmoid function to get the binary output. Compare with RNN, LSTM has a more complex structure named LSTM cell which adds its hidden layer. The former output is h_{t-1}, and the cell's former state is C_{t-1}. Compared with

RNN that has only one transmission state h_t, LSTM has two transmission states, including one cell state, and one hidden state. h_t in RNN corresponds to C_t in LSTM; the cell state changes slowly; and the output is usually C_{t-1} from the previous state plus some value. The hidden state can vary greatly from node to node. This feature behaves like the traffic data, which is highly context-related, and it takes time to absorb the change, and changes gradually.

4.2.3. Attention Mechanism

In the applications, LSTM indicates that such approaches for prediction are suitable and can achieve better performances than traditional models. However, in the neural network's training process, the more parameters of the model, the more meaningful the model interpretation, and the more information the model stores, which will also bring information overload problem. When the traffic conditions considerably change, the prediction results are often unsatisfactory. Attention Mechanism in neural networks is a resource allocation scheme to allocate computing resources to more critical tasks and solve the problem of information overload when computing capacity is limited. The attention mechanism can preset the model to assign different weights to each input feature to extract more critical information from the highly related features. This mechanism can make the model more accurate. Meanwhile, it will not bring more overhead to the calculation and storage of the model. In this research, the attention mechanism is proposed to address the drawbacks of LSTM, in which the attention mechanism is over the output layer of each LSTM unit. The attention mechanism substitutes the traditional recurrent way to construct the depth of LSTM.

In the attention mechanism, the key point is the resulting probability vector which is called the memory distribution and it is presented as follows:

$$P_i = p(\mathrm{z} = \mathrm{i}|\mathrm{X}, \mathrm{q}) = softmax\big(s(x_i\ , q)\big) = \frac{\exp\big(s(x_i, q)\big)}{\Sigma_{j=1}^{N} \exp\big(s(x_i, q)\big)} \qquad (10.)$$

where,

z: is the defined attention variable, and $\mathrm{z} = \mathrm{i}$ means that the attention variable is the variable i,

i$\in [1, N]$;

X: can be thought of as information store;

q: represents the query vector.

CHAPTER 5: TTP MODEL VALIDATION

Chapter 5 discusses the validation of the developed TTP models based on the dataset gathered and presented in Chapter 3. For the RF and LSTM_AM models, the data training steps are described to determine the optimal parameters. Potential features include, but are not limited to, the TOD, DOW, month, year, weather conditions, and segment characteristics. The selection of hyper-parameters, such as the number of hidden units and iterations, is also further discussed.

5.1. RF Algorithm

In order to optimize the performance of the proposed method, it is essential to test the effect of different combinations of parameters on the model performance. Based on previous research (Zhang and Haghani, 2015), the parameters that can be tested and optimized include, but are not limited to, Max_features, N_estimators (number of trees), and Min_sample-leaf. In general, these parameters need to be optimized in this model training process.

The grid search method is the most widely used one, and therefore, it is selected and used in the tuning process. In this experiment, 70% of the traffic data is used as training data, and 30% of the data is used as testing data. The RF model is fitted with various numbers of trees (*N_estimators* ranging from 1 to 500), *Min_sample-leaf* (ranging from 5 to 50), and Max-feature (which is calculated by using the equation, $m = INT(log_2 M + 1)$). For example, when M=35, $m = INT(log_2 35 + 1) = 6$.

From Figures 5.1, 5.2 and Table 5.1, the results show that the optimal *N_estimator* is achieved at 50 using the *Max_feature* with the log2 method, and the values of MAPEs stay nearly the same after that optimal point. In statistics, overfitting is the product of an analysis that corresponds too precisely to the sample set of data. Therefore, it may fail to fit additional data or predict future observations reliably, which is the main weakness of ensemble learning approaches. In other words, the MAPE does not decrease when the *N_estimator* reaches the optimal point in the tree-base model. It is also essential to consider the balance between prediction accuracy and computational efficiency, and it is evident when the model complexity increases, and the computational time increases significantly.

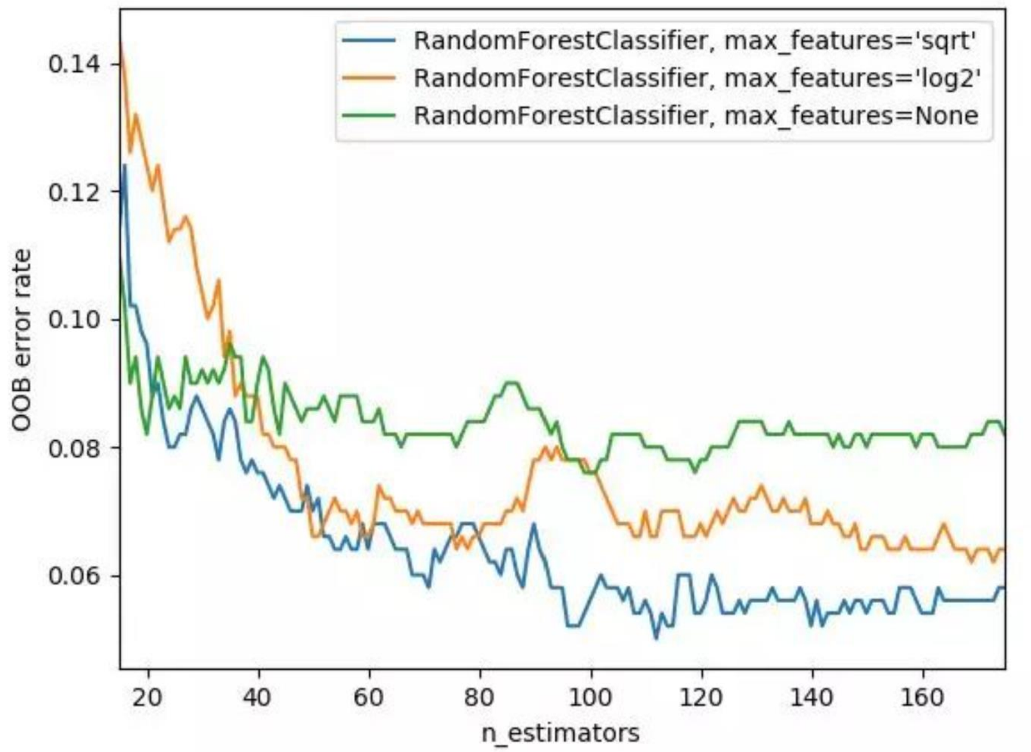

Figure 5.1 *Max_features* and the number of tree tuning process

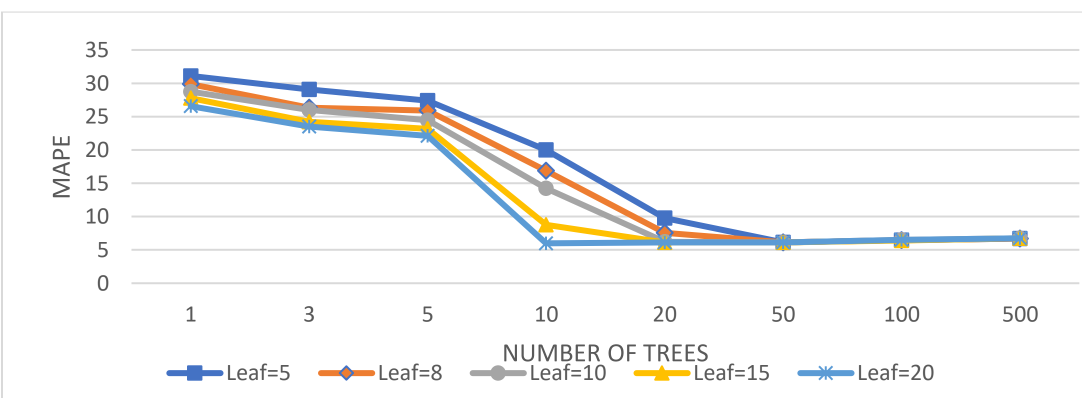

Figure 5.2 RF TTP model performance with *Max_feature* =6

Table 5.1 The MAPE of the combination of parameters

Number of trees	**Leaf=5**	**Leaf=10**	**Leaf=20**	**Leaf=30**	**Leaf=50**
1	31.11	29.87	26.56	26.01	26.74
3	29.05	26.34	23.52	22.46	23.59
5	27.38	25.9	22.09	21.28	22.24
10	19.98	16.87	6.13	6.01	6.26
20	9.78	7.56	6.1	5.99	6.05
50	6.13	6.14	6.12	5.97	5.99
100	6.46	6.48	6.51	6.42	6.54
500	6.7	6.72	6.73	6.6	6.72

To measure the effectiveness of different TTP algorithms, the MAPEs are computed for three different observation segments (in which A, B, C are three observation segments along the selected study freeway as shown in Figure 4.4) with varying horizons of prediction that range from 15 minutes to 60 minutes. According to the comparison as shown in Table 5.2 and Figure 5.4, the performance of RF is better than the XGB, particularly when the horizon of the prediction time is extended. The MAPE of the RF model is noticeably smaller than XGB when the horizon is long enough (i.e., longer than 45 min).

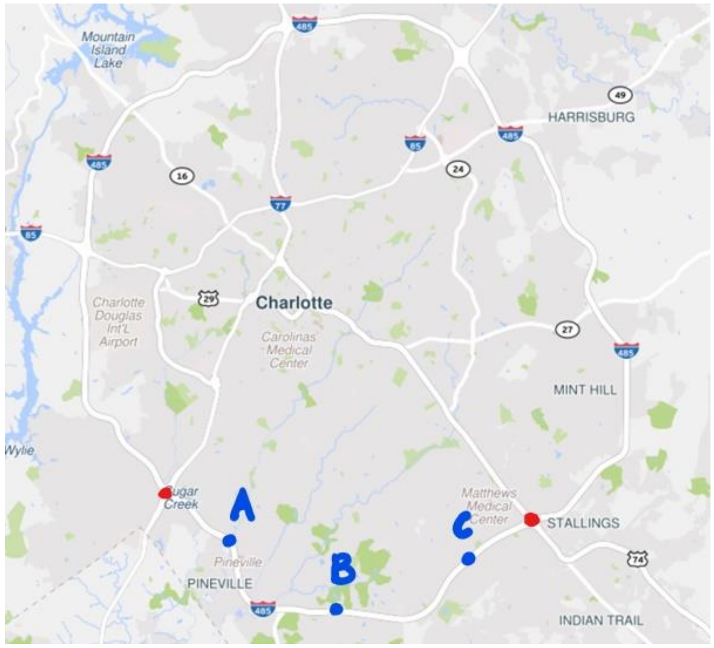

Figure 5.3 Observation points in the selected segment

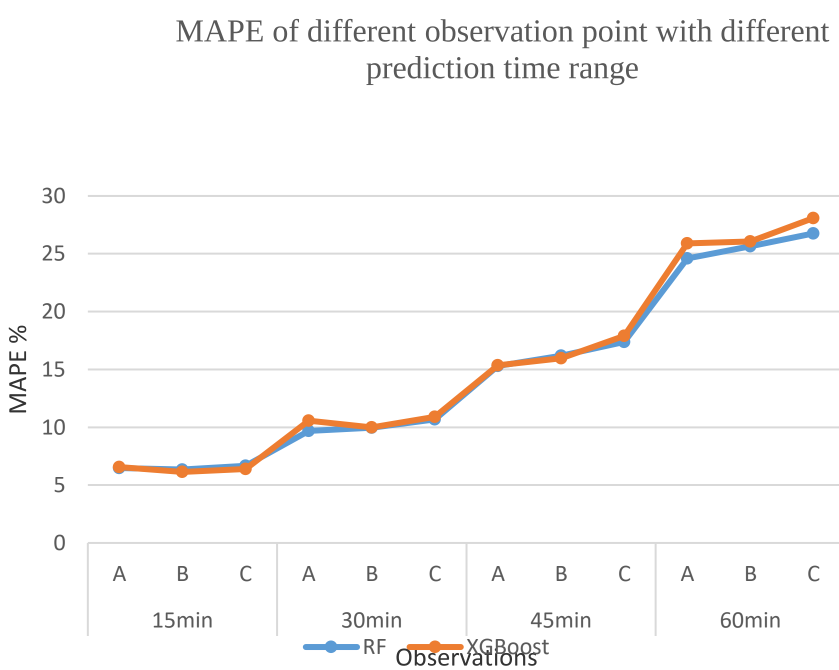

Figure 5.4 RF TTP model performance comparison

Table 5.2 RF model comparison of different prediction methods

MAPE (%) of different observation points with different prediction time range												
Models	**15min**			**30min**			**45min**			**60min**		
	A	B	C	A	B	C	A	B	C	A	B	C
RF	6.49	6.15	6.39	9.69	9.97	10.67	15.29	16.19	17.37	24.59	25.66	26.76
XGB	6.57	6.14	6.39	10.58	9.98	10.89	15.35	15.98	17.90	25.90	26.06	28.09

In the feature selection process, the RF model is employed to rank the relative importance using the original dataset. The features that have the importance of more than 0.1% are selected in the model training process. In this research, 23 features are selected from the original 35 features (with the least important feature being the length of the road segment at 0.17%). Table 5.3 presents the relative importance for each selected feature and its ranks in the proposed RF model. The model results show that the variable T_{t-1}(i.e., travel time 15 minutes before) has an immense contribution (i.e., 34.85%) to the prediction result. The immediate previous traffic condition has the most critical impact on the traffic condition in the future, which is consistent with a previous study conducted in 2015 (Zhang and Haghan, 2015). TOD is the second-highest important feature with an RIV of 30.12%, which is also under the expectation. T_{t-w} is the fourth-highest ranked variable with a 9.87% RIV, which can be interpreted as a highly similar pattern of traffic times between two adjacent weeks.

The results in Table 5.3 also show that the spatial impact is less than the time impact since the RIVs of all the spatial variables are less than 1% (except the variable road ID with an RIV of 2.28%). The features T_{t-1}^{i-1}, T_{t-1}^{i-2} (i.e., the travel time of the nearest and second nearest upstream segments 15 minutes before) with the RIVs of 0.31% and 0.42%, respectively. In the meantime, the features T_{t-1}^{i+1}, T_{t-1}^{i+2} (the travel time of the nearest and second nearest downstream segments) with the RIVs of 0.35% and 0.61%. Concerning the travel time change value, the RIVs of the nearest and second nearest downstream segments are both 0.29%, and the RIVs of the nearest and second nearest upstream segments are 0.79% and 0.37%, respectively. These experimental results indicate that the impact of the downstream segments is more significant than those of upstream segments. In this regard, the spatial characteristics of the roadways clearly help explain that when the bottleneck occurs at the downstream segments, the upstream will be impacted very soon.

Table 5.3 RIV of each feature in the TTP model

Variable	**Definition**	**Relative importance (%)**	**Attribute**
ID	Road segment ID	2.28	7
L	Length of the road segment	0.17	23
Speed	Space Mean Speed	10.59	3
TOD	Time of day is indexed from 1 to 96, which represent the time from 0:00-24:00 by every 15-minute timestep	30.12	2
DOW	Day of week is indexed from 1 to 7, which represent from Monday through Sunday	2.84	5
Month	The month is indexed 1 to 12, which represent from January to December	1.59	8
Weather	Weather is indexed from 1 to 3, which represent normal, rain and snow/ice/fog	2.63	6
T_{t-1}	The travel time at the prediction segment 15 minutes before	34.85	1
T_{t-2}	The travel time at the prediction segment 30 minutes before	0.57	11
$\mathrm{T}_{\mathrm{t-3}}$	The travel time at the prediction segment 45 minutes before	0.28	18
T_{t-w}	The travel time at prediction segment one week before	9.87	4
ΔT_{t-1}	The travel time change value at $\mathrm{T}_{\mathrm{t-1}}$	0.24	19
ΔT_{t-2}	The travel time change value at $\mathrm{T}_{\mathrm{t-2}}$	0.20	21
ΔT_{t-3}	The travel time change value at $\mathrm{T}_{\mathrm{t-3}}$	0.18	22
ΔT_{t-w}	The travel time change value at $\mathrm{T}_{\mathrm{t-w}}$	0.22	20
T_{t-1}^{i-1}	The travel time of the nearest upstream road segment 15 minutes before	0.31	15
T_{t-1}^{i-2}	The travel time of the second nearest upstream road segment 15 minutes before	0.42	12
ΔT_{t-1}^{i-1}	The travel time change value at the nearest upstream road segment 15 minutes before	0.29	16
ΔT_{t-1}^{i-2}	The travel time change value at the second nearest upstream road segment 15 minutes before	0.29	16
T_{t-1}^{i+1}	The travel time of the nearest downstream road segment 15 minutes before	0.35	14
T_{t-1}^{i+2}	The travel time of the second nearest downstream road segment 15 minutes before	0.61	10
ΔT_{t-1}^{i+1}	The travel time change value at the nearest downstream road segment 15 minutes before	0.79	9
ΔT_{t-1}^{i+2}	The travel time change value at the second nearest downstream road segment 15 minutes before	0.37	13

5.2. LSTM_AM Algorithm

LSTM_AM algorithm contains two training steps. The first one is the training of LSTM, and the second is the training of the ODC matrix (Zhao et al., 2017). To represent the highly nonlinear function of the model, the neural networks always have many levels of non-linearities. Hinton and Salakhutdinov (2006) introduced a training method named greedy layer-wise unsupervised learning algorithm. Model training is the parameters' determination process in the predetermined or designed model structure. In the development of LSTM prediction model, the dimension of the hidden layer and the number of iterations are the two most critical hyper parameters, which must be predetermined separately from the training process of the other parameters. The sample data set is divided into the training set, including 70% sample data, and the validation set, which consists of 30% sample data. The adjustment relies on the training set since both the dimension of the hidden layer and the number of iterations are determined by the prediction performance on the test set and evaluation of the cost function on the validation set.

The LSTM network is trained layer by layer. At the first step, the weight matrices and bias vectors ($W^{(i)}, W^{(f)}, W^{(o)}, W^{(c)}, U^{(i)}, U^{(f)}, U^{(o)}, U^{(c)}$) are initialized and assigned the weight randomly. Then, the parameters are trained by using a backward propagation method with gradient-based optimization, which can be solved by minimizing the cost function (Zhao et al., 2017). Secondly, the ODC matrix needs to be predetermined at different time intervals depending on the sample dataset. No predetermined time window size is to be required since LSTM can automatically calculate the optimal time lags (Ma et al., 2015). Since the sampling frequency is 15 minutes in the dataset, the proposed model predicts travel time at 30 minutes, 45 minutes, and 60 minutes intervals (i.e., the number of layers in the neural network is set as 2, 3, 5, and 6, respectively).

5.2.1. LSTM_AM Training Steps

The following steps describe the training process of the LSTM_AM model. Furthermore, the model performance with different epochs in the training process is shown in Figure 5.5.

Step 1: Check the data and convert the data to the standard format.

Step 2: Shuffle the data, and then split the data into training and testing.

Step 3: Shuffle the data before feeding it into the LSTM_AM model.

Step 4: Train the data using the LSTM_AM model.

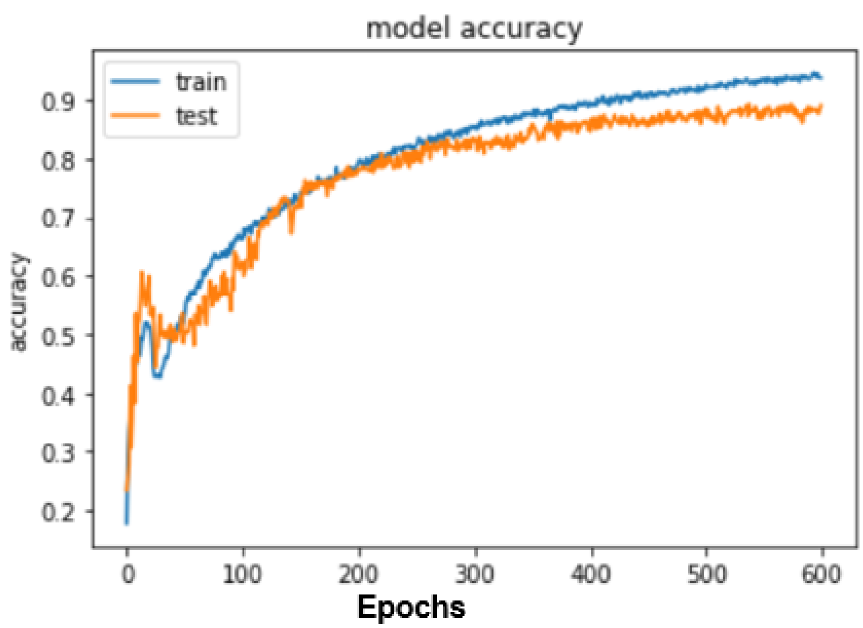

Figure 5.5 Model performance with different epochs in the training process

To measure the effectiveness of the proposed LSTM_AM TTP algorithm and other methods, the MAPEs are computed. The comparison results as shown in Table 5.4 and Figure 5.5 show that the proposed LSTM_AM model has a significantly improved performance over the DT and LSTM, especially when the prediction horizon is long. The MAPEs of the LSTM-based model with attention mechanism is smaller than the other methods when the horizon is long enough (e.g., longer than 45 minutes). LSTM indicates a better performance compared with DT. When the prediction horizon is 15 minutes, the LSTM algorithm performance is as good as the LSTM_AM. However, when the prediction horizon is longer than 30 minutes, the MAPEs increase significantly. Thus, the results demonstrate that both the DT and LSTM methods lack precision when dealing with long-term prediction problems. The possible reason is that the causal relation of the time step dimension is partially ignored by the tree structure algorithm when fusing the data into the models. The experimental results from the sample dataset indicate that the proposed LSTM_AM model performance is better than the existing LSTM and other baseline methods, which primarily can achieve a higher accuracy in the long-range prediction horizon.

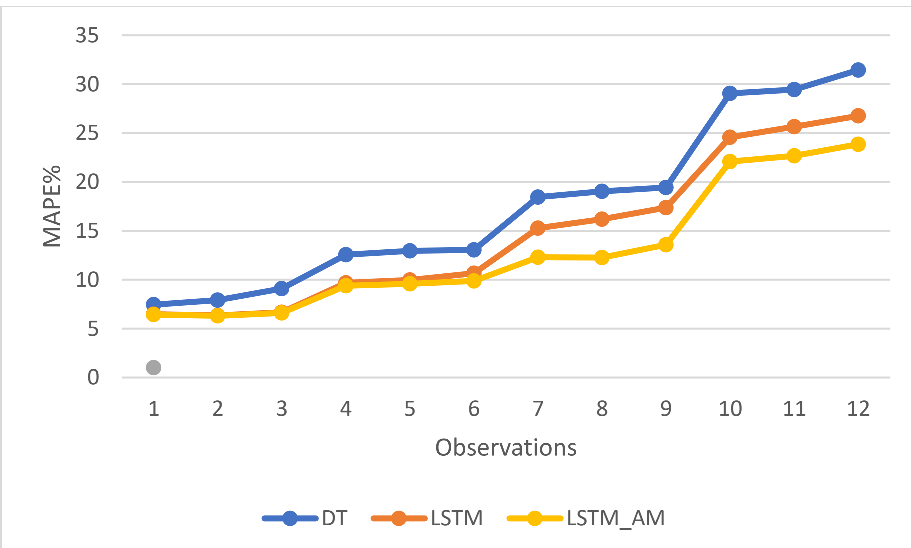

Figure 5.6 LSTM_AM TTP model performance comparison

Table 5.4 LSTM model comparison of different prediction methods

MAPE (%) of varying observation points with different prediction time range												
Models	15min			30min			45min			60min		
	A	B	C	A	B	C	A	B	C	A	B	C
DT	7.45	7.9	9.08	12.56	12.97	13.05	18.45	19.04	19.45	29.05	29.45	31.45
LSTM	6.49	6.35	6.67	9.69	9.97	10.67	15.29	16.19	17.37	24.59	25.66	26.76
LSTM_AM	6.44	6.31	6.6	9.4	9.59	9.87	12.29	12.28	13.59	22.08	22.69	23.86

After the development of the proposed LSTM_AM model, a unified comparison of the five methods as shown in Figure 5.7 is conducted by using the same group of sample data that was applied in the development of the two groups of models. The 15-minute prediction group comparison is also conducted and presented in Figure 5.8. The results indicate that all proposed TTP models predicting at a 15 minute interval show a better performance over the other time horizons. Besides, the RF model has the best prediction performance with an average MAPE of 6.34% on the 15 minute prediction horizon (LSTM_AM is 6.45%). The LSTM_AM model has the best performance in all other predictions horizons (30 min, 45 min, and 60 min). In practice, they can be applied to their preferred prediction horizons.

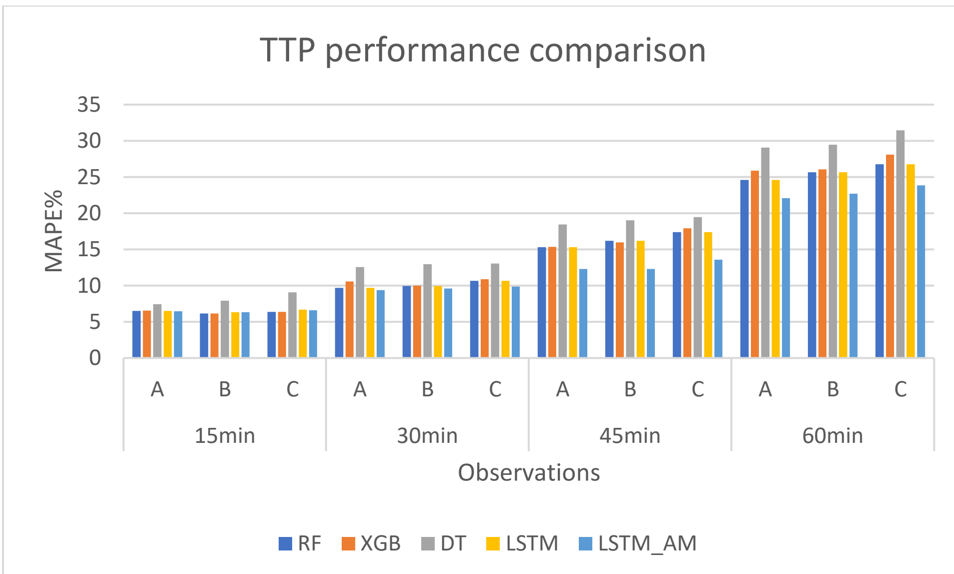

Figure 5.7 TTP performance comparison at different prediction intervals

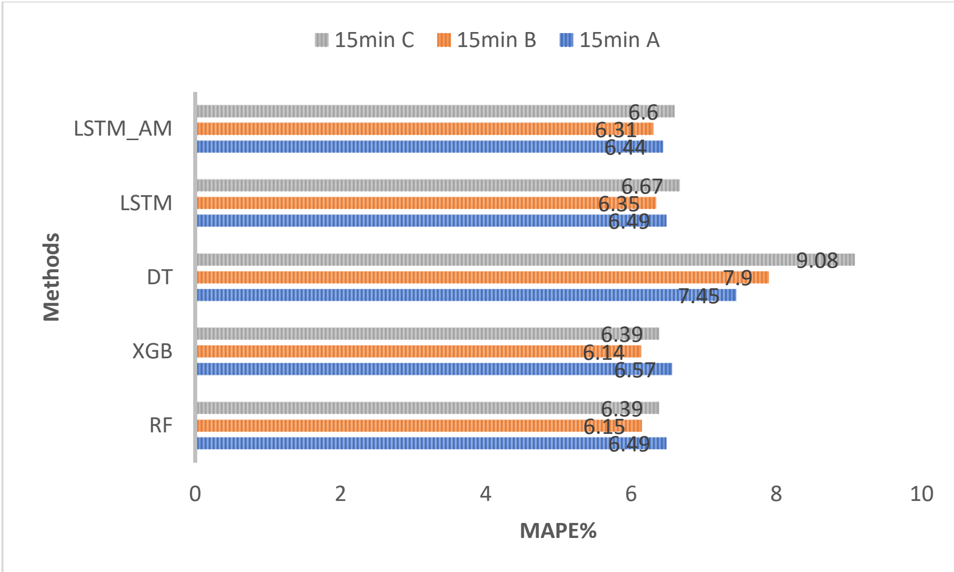

Figure 5.8 TTP performance comparison in the 15 min prediction horizon

The non-parametric models tend to be more efficient and, therefore, have a more advanced model structure. It is mentioned that the efficiency of data-driven approaches, in general, is poor and not suitable for real-time applications (Oh et al., 2015). The artificial neural network is one of the most popular methods in the literature of TTP, perhaps due to their ability to capture complex relationships in large data sets and their more efficient calculations. Due to the introduction of the attention mechanism algorithm, the LSTM_AM algorithm is more efficient in capturing adequate information. In Table 5.5, the operation efficiency of various methods is also compared and shown. LSTM_AM is more realistic in the real-time application of TTP due to the high efficiency of calculation. Note that the 30-minute travel time is predicted based on the calculation and use of the previous 15-minute travel time. The computation time increases in all proposed models except in RF, which could be caused by the random selection process. Furthermore, with the introduction of the random select algorithm, the RF is also more efficient than other ensemble learning methods.

Table 5.5 The computation time comparison of different prediction methods

Models	15min			30min		
	A	B	C	A	B	C
LSTM_AM	11.2s	12.5s	15.1s	12.4s	12.9s	14.5s
LSTM	34.5s	35.1s	39.0s	38.5s	40.2s	45.8s
DT	112s	117s	127s	145s	137s	125s
RF	45.3s	44.7s	48.9s	39.6s	43.1s	42.2s

CHAPTER 6: PREDICTION RESULTS ANALYSIS

Chapter 6 provides the evaluation and analysis of the TTP modeling results. The statistical index MAPE is used to measure the prediction error.

6.1. Modeling Results Analysis

In machine learning, overfitting typically occurs when the model corresponds perfectly to the sample set of data. Therefore, the model may fail to fit additional data or predict future observations reliably. RF is an ensemble of DTs. The single DT is sensitive to data variations, which can overfit to noise in the data. While in the RF model, as the N_estimater increases, the tendency of overfitting decreases. The RF is not prone to overfitting and is very noise-resistant due to the bagging and random feature selection process. However, it can still be improved, and to avoid overfitting in RF, the hyper-parameters of the algorithm should be tuned very carefully.

TTP is based on accurate modeling of the complex nonlinear spatiotemporal traffic dynamics in the real world (Ran et al., 2019). The accuracy and interpretability of models are two major concerns. In general, RNN is more like a complex black-box model aiming for achieving the accuracy versus less accurate but more interpretable for traditional models such as linear regression (Choi et al., 2016). In recent years, the increased congestion on freeways has led to increasing uncertainty, making it more challenging for the TTP model to achieve preset prediction accuracy. In this research, a systematic machine learning solution is developed for short-term TTP. A feature select preparation step is developed to overcome the drawbacks in the existing methods with the incorporation of many spatial and temporal characteristics that may affect travel time. TTP accuracy can be significantly improved by reducing the time-lag problems when both spatial and temporal characteristics are considered (Lee et al., 2020). In statistical theory, the attention mechanism tends to make the model more efficient and accurate.

Meanwhile, it will not bring more overhead to the calculation and storage of the model. Therefore, the attention mechanism can be introduced to focus on the information that is more critical to the current prediction task among numerous inputs, reduce attention to other details, or even filter out irrelevant information. This way, one can solve the problem of information overload and improve the efficiency and accuracy of task processing. This is similar to the visual attention mechanism of human beings. By scanning the global image, one can obtain the target area that needs to be focused on and then devote more attention to this area to get more details related to the target while ignoring other irrelevant information. Due to the ability to concentrate on the compelling parts of features adaptively, this approach has been successful in image classification (Mnih et al., 2014), neural machine translation (Luong et al., 2015), multimedia recommendation (Chen et al., 2017), and some other fields.

The proposed RF and LSTM_AM models are developed to estimate and forecast the freeway travel time, and the results show that the prediction accuracy and model reliability are improved significantly. Most existing machine learning models can deal with the nonlinear pattern of travel time but suffer from low accuracy when the prediction range gets longer (Zhang et al., 2020). Results indicate that the LSTM_AM model can provide reliable prediction results for the 15 minutes to 60 minutes time ranges. The relative importance of the features shows that the travel time one step ahead (i.e., 15 minutes before) contributes the most to the predicted travel time. Features (such as the TOW, DOW, the travel time at prediction segment one week before, and weather) also have higher RIVs in the model than other features. Adding up the most critical six variables' RIVs (i.e., T_{t-1}, TOD, Speed, T_{t-w}, DOW, and Weather) in Table 5.3 is as high as 90.90%, which means that these six selected variables include most of the information needed in the TTP. Table 5.3 also shows that the time features (such as T_{t-1}, TOD, T_{t-w}, and DOW) has a significantly higher RIV than the other features (such as weather, road ID, length, and speed). It can be seen from Figure 5.5, the proposed RF and LSTM_AM methods have considerable advantages over the comparable approaches in short-time TTP. The results from the comparison of LSTM and tree-based models also reveal the complexity and difficulty in the optimization for machine learning prediction models. LSTM has an automatic optimizer to optimize the hyperparameters (node, layer, and batch-size). However, for the tree-based models, one needs to set and optimize the hyperparameters and learning rate, which could be a significant amount of work for optimization considering changes of all the hyperparameters. In this research, a set of appropriate default values is employed for these parameters based on the previous research to simplify the comparison.

In summary, prediction results show that the proposed LSTM_AM model outperforms LSTM with a 0.053% lower on average in MAPE (i.e., 6.450% vs 6.503%), however, with a higher MAPE than RF and XGB in 15-minute prediction horizon (i.e., 6.343% [RF], and 6.366% [XGB]). The proposed RF model has the best prediction performance with an average MAPE of 6.34% on 15-minute prediction horizon, and the LSTM_AM model has the best performance in all other prediction horizons (30min, 45min, and 60min). The proposed RF model is expected to have better compatibility (benefit of avoiding overfitting), which means that it may have better performance in the new dataset. However, LSTM_AM has the best computational efficiency. In practice, they can be applied to their preferred prediction horizons.

6.2. The Effect of Prediction Horizon

For different prediction horizons, the four most important variables are the same, and they are travel time at prediction segment 15 minutes before, TOD, speed, and travel time one week before. As expected, the travel time of the current period has the greatest influence on the travel time of the next period.

Table 6.1 Relative importance for different prediction horizons

Variable	Definition	**15 min prediction horizon**	**30 min prediction horizon**	**45 min prediction horizon**
ID	Road segment ID	8	7	9
L	Length of the road segment	23	23	16
Speed	Space Mean Speed	3	3	3
TOD	Time of day indexed from 1 to 96, which represent the time from 0:00-24:00 by every 15-minute timestep	2	2	2
DOW	Day of week indexed from 1 to 7, which represent from Monday through Sunday	6	5	7
Month	Month is indexed 1 to 12, which represent from January to December	10	8	12
Weather	Weather indexed from 1 to 3, which represent normal, rain and snow/ice/fog	5	6	8
T_{t-1}	The travel time at the prediction segment 15 minutes before	1	1	1
T_{t-2}	The travel time at the prediction segment 30 minutes before	7	11	14
T_{t-3}	The travel time at the prediction segment 45 minutes before	19	18	23
T_{t-w}	The travel time at the prediction segment one week before	4	4	4
ΔT_{t-1}	The travel time change value at T_{t-1}	16	19	17
ΔT_{t-2}	The travel time change value at T_{t-2}	20	21	22
ΔT_{t-3}	The travel time change value at T_{t-3}	22	22	20
ΔT_{t-w}	The travel time change value at T_{t-w}	21	20	18
T_{t-1}^{i-1}	The travel time of the nearest upstream road segment 15 minutes before	14	15	19
T_{t-1}^{i-2}	The travel time of the second nearest upstream road segment 15 minutes before	11	12	10
ΔT_{t-1}^{i-1}	The travel time change value at the nearest upstream road segment 15 minutes before	18	16	13
ΔT_{t-1}^{i-2}	The travel time change value at the second nearest upstream road segment 15 minutes before	17	16	21
T_{t-1}^{i+1}	The travel time of the nearest downstream road segment 15 minutes before	13	14	15
T_{t-1}^{i+2}	The travel time of the second nearest downstream road segment 15 minutes before	9	10	6
ΔT_{t-1}^{i+1}	The travel time change value at the nearest downstream road segment 15 minutes before	12	9	5
ΔT_{t-1}^{i+2}	The travel time change value at the second nearest downstream road segment 15 minutes before	15	13	11

Since the most important relative feature is the same for different prediction horizons, the partial dependence function graphs between predicted travel time and actual travel time in the current period are shown in Figure 6.1. From Figure 6.1, it can be found that current travel time has a highly linear relationship with the predicted travel time; however, the curve behaves differently for different prediction horizons. Furthermore, when the prediction horizon increases (from 15 to 45 min), the change rate of the curve gradually decreases, which demonstrates that travel time in the current period has less impact on the TTP. It indicates that the model's predicted performance decreases as the prediction horizon increases.

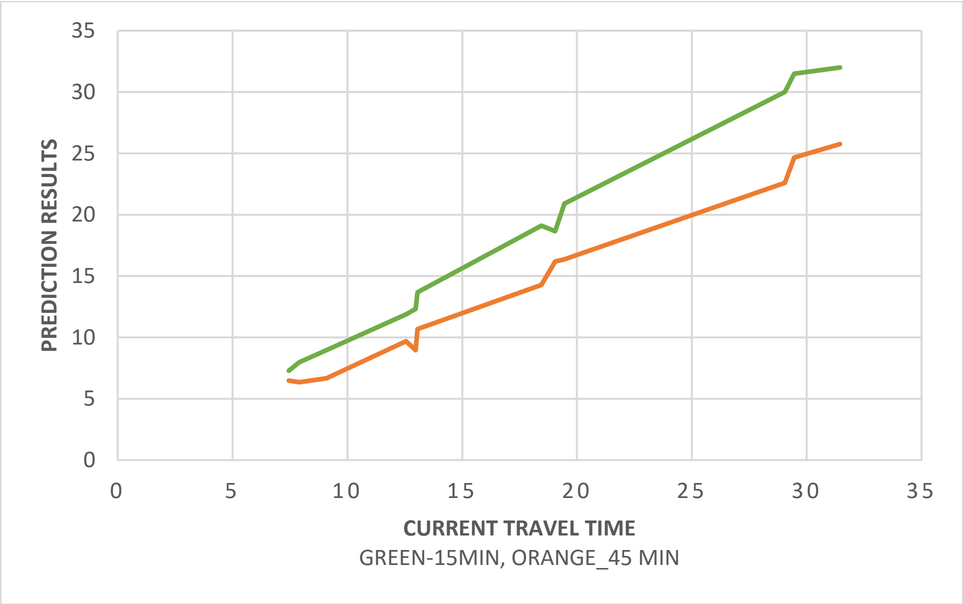

Figure 6.1 Partial dependence function graph for different prediction horizons

The prediction horizon effect is consistent with the most commonly used statistical methods, i.e., linear regression (LR). In the LR model, the dependent variable is a linear function of the independent (explanatory) variables. LR demonstrates that travel time in the close future can be estimated based on current travel time. The coefficients of regression lines are functions of current time and prediction horizons that are shown in the function below.

$$T(t+h) = \alpha(t,h) + \beta\,(t,h) \times T(t)$$

The LR method shows that the predicted travel time at t+h $T(t+h)$ can be presented by a linear function of current travel time (t) and prediction horizon (h). One can see that the intercept α and slope β are both a function of h. It can also be seen from this graph that when the prediction horizon increases, the slope decreases, which is consistent with the effect as expected.

CHAPTER 7: SUMMARY AND CONCLUSIONS

Chapter 7 concludes with a summary of the TTP results. Suggestions for future research are also provided.

7.1. Conclusions

RF and LSTM_AM TTP models were developed and applied using the RITIS dataset with selected variables over different prediction time horizons (i.e., 15 min, 30 min, 45 min, and 60 min). DT was selected one of the traditional algorithms, and enhanced ensemble learning methods XGB and LSTM were also chosen as the baseline methods to compare with the proposed TTP models. The results indicated that all the proposed TTP models have better performance and higher prediction accuracy in predicting travel times at the 15 minute interval over the other time horizons. Besides, the RF model had the best prediction performance with an average MAPE of 6.34% in the 15 minutes prediction horizon, and the LSTM_AM model had the best performance in all other prediction horizons (i.e., 30min, 45min, and 60min). In practice, they can be applied to their preferred prediction horizons.

Most existing machine learning models can capture the nonlinear pattern of travel time but suffer from over-fitting. The tree-based ensemble methods have been widely used in the field of prediction due to their benefits of avoiding overfitting. Combining a simple tree into a forest always produces high prediction accuracy (Zhang and Haghani, 2015). In this research, the RF method was applied to analyze and model freeway travel time to improve the prediction accuracy and interpretability. Study results indicated that the RF model has considerable advantages in freeway TTP. The performance evaluation results also showed that the RF-based model can have better predictions in terms of prediction accuracy in the short time prediction horizon (i.e., 15 minutes). However, when the prediction horizons become longer than 30 minutes, the errors increase significantly in other methods. Different from other machine learning methods, RF methods provide interpretable results with varying types of predictor variables. RF can also handle data with very high dimensions (i.e., many features) without feature selection (because feature subsets are randomly selected) and identify which features are more critical after the training process. Furthermore, it effectively estimates missing data and maintains accuracy when a significant proportion of the data is missing. In summary, the proposed RF TTP method has considerable advantages over the other tree-based approaches.

The performance evaluation results showed that the LSTM_AM model performs better in terms of both prediction accuracy and efficiency in all ranges of short-term TTP. In practice, efficiency is as crucial as accuracy in the application. As the level of the information detail increases, the parameter tunings are expected to require more computational efforts. LSTM_AM procedures require a relatively shorter processing time (even for an extensive network) than other machine learning methods. Compared with other proposed methods, the LSTM_AM model consumes less time. In addition, in the verification of prediction accuracy, the performance of XGB and DT algorithms has poor performance in training time and is less practical in real-time prediction. LSTM_AM model can also handle hundreds of input variables without variable deletion. It is noted that especially LSTM_AM can still maintain considerable prediction accuracy when a large proportion of the data are missing. In this research, the incomplete data was calculated according to its missing type, which effectively improves the quality of the data, enhances the usability of sample data, and enhances the model's accuracy to a certain extent. In summary, through the validation and comparative analysis of the LSTM_AM, it is found that this method has outstanding performance and applicability for short-term TTP on freeways.

In summary, the RF model and a new TTP method LSTM_AM have been developed by using the RITIS dataset on the Charlotte freeways. Three conventional machine learning models (i.e., DT, XGBoost, and LSTM) have been developed by using the same dataset to perform the model comparison. The prediction model is tested on the RITIS dataset, and a large set of time, spatial, and weather-related variables was generated (collected) as additional input features (i.e., T_{t-1}, T_{t-2} …, $T_{t-\mathrm{w}}$, ΔT_{t-1},…, ΔT_{t-w} , T_{t-1}^{i-1}…). $T_{t-\mathrm{w}}$ was first introduced as an input feature in TTP and showed that it is very useful to improve the model accuracy. LSTM_AM had a better ability to model the traffic dynamics in road networks as they can model long-term dependence in time series and extract features from traffic data with recurrent feedback to obtain long-term accuracy. For a practical 15-minute prediction interval, the predicted travel time values were accurate to within 1 minute of the actual values on most of the routes. In addition, the prediction model also performed quite well even if the prediction interval was large (e.g., at the 60-minute interval). The maximum error in such a case was around 5 minutes on a route of 18.6 miles.

7.2. Future Research Directions

The practice of RF algorithm and LSTM methods in the TTP area is still very limited. The future focus of the research would be hybrid models (combination models), which can combine several models of the same or different types of prediction models to enhance the model performance and prediction. The proposed RF method can be combined with other tree-based methods or another type of machine learning method in the preprocessing step or prediction step. The LSTM method can be combined with other tree-based methods or another kind of machine learning method in the preprocessing step or prediction step. Experimental results showed that the combination methods have better prediction than using a technique alone (Li et al., 2009). As the combination model method has been proved superior in prediction accuracy, this should be given careful consideration in the future.

In order to determine whether the prediction results are region-specific, continued research is also needed to replicate this study in other types of road categories that exhibit different characteristics. More results need to be achieved so as to compare all methods, which may help further demonstrate and confirm that the proposed methods have better predictive accuracy in short-term TTP. More variables related to traffic characteristics and surroundings such as traffic volumes, speed limit, and sun angles could be integrated into analysis if available.

In addition, a new data structure can be generated and a time-specific model can be trained (i.e., creating 30 min interval data to predict 30 min travel time) to test the prediction accuracy and efficiency by comparing them with the current prediction method. It is essential to consider the severe weather's impact on the travel time pattern. In practice, one also needs to reduce the effect of extreme values that create biased predictions. It is known that reducing the effect of exogenous factors can improve the TTP accuracy in the model validation, such as incidents and weather. However, the TTP models under special weather conditions, such as the travel time pattern under heavy rain, snow, fog, and ice, are valuable. Furthermore, low visibility on freeways provides challenges for proactive traffic safety management and TTP under fog conditions. Few studies have developed prediction models with a focus on visibility on freeways in a short-term time horizon. Therefore, future studies should also establish TTP models considering the visibility of freeways at a short-term time horizon.

REFERENCES

1. Ahmed, M.S. and Cook, A.R., 1979. Analysis of freeway traffic time-series data by using Box-Jenkins techniques. Transportation Research Record, 722, 1-9.
2. Bengio, Y., Lamblin, P., Popovici, D. and Larochelle, H., 2007. Greedy layer-wise training of deep networks. In Advances in neural information processing systems (pp. 153-160).
3. Bengio, Y., Simard, P. and Frasconi, P., 1994. Learning long-term dependencies with gradient descent is difficult. IEEE transactions on neural networks, 5(2), pp.157-166.
4. Breiman, L., 2001. Random forests, machine learning 45. J. Clin. Microbiol, 2(30), 199-228.
5. Chen, J.; Zhang, H.; He, X.; Liu, W.; Liu, W.; Chua, T.S. Attentive Collaborative Filtering: Multimedia Recommendation with Item- and Component-Level Attention. In Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, Tokyo, Japan, 7–11 August 2017; pp. 335–344.
6. Chen, Z. and Fan, W., 2019. Data analytics approach for travel time reliability pattern analysis and prediction. Journal of Modern Transportation, 27(4), pp.250-265.
7. Chen, T. and Guestrin, C.: 'XGB: A Scalable Tree Boosting System', in, Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, (2016)

8. Choi, E., Bahadori, M.T., Kulas, J.A., Schuetz, A., Stewart, W.F. and Sun, J., 2016. Retain: An interpreTablepredictive model for healthcare using reverse time attention mechanism. arXiv preprint arXiv:1608.05745.

9. Dharia, A., & Adeli, H., 2003. Neural network model for rapid forecasting of freeway link travel time. Engineering Applications of Artificial Intelligence, 16(7-8), 607-613.

10. Dion, F., Rakha, H., & Kang, Y. S., 2004. Comparison of delay estimates at under-saturated and over-saturated pre-timed signalized intersections. Transportation Research Part B: Methodological, 38(2), 99-122.

11. Dong X, T Lei, S Jin, and Z Hou. 2018. Short-term traffic flow prediction based on xgboost. 2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS). Dali, China, May 24-27.

12. Du, L., Peeta, S., & Kim, Y. H., 2012. An adaptive information fusion model to predict the short-term link travel time distribution in dynamic traffic networks. Transportation Research Part B: Methodological, 46(1), 235-252.

13. Duan, Y., Yisheng, L.V. and Wang, F.Y., 2016, November. TTP with LSTM neural network. In 2016 IEEE 19th international conference on intelligent transportation systems (ITSC) (pp. 1053-1058). IEEE.

14. Fan, S. K. S., Su, C. J., Nien, H. T., Tsai, P. F., & Cheng, C. Y., 2018. Using machine learning and big data approaches to predict travel time based on historical and real-time data from Taiwan electronic toll collection. Soft Computing, 22(17), 5707-5718.

15. Greenhalgh, J. and Mirmehdi, M., 2012. Real-time detection and recognition of road traffic signs. IEEE transactions on intelligent transportation systems, 13(4), pp.1498-1506.

16. Gupta, B., Awasthi, S., Gupta, R., Ram, L., Kumar, P., Prasad, B. R., & Agarwal, S., 2018. Taxi TTP using ensemble-based random forest and gradient boosting model. In Advances in Big Data and Cloud Computing (pp. 63-78). Springer, Singapore.

17. Hamner, B., 2010. 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.

18. He, H. S., Keane, R. E., & Iverson, L. R., 2008. Forest landscape models, a tool for understanding the effect of the large-scale and long-term landscape processes. Forest Ecology and Management. 254: 371-374., 254.

19. Hinton, G.E. and Salakhutdinov, R.R., 2006. Reducing the dimensionality of data with neural networks. science, 313(5786), pp.504-507.

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

21. Jiang, X., & Adeli, H., 2004. Wavelet packet‐autocorrelation function method for traffic flow pattern analysis. Computer ‐ Aided Civil and Infrastructure Engineering, 19(5), 324-337.

22. Koetse, M.J. and Rietveld, P., 2007, September. Climate change, adverse weather conditions, and transport: A literature survey. In Proceedings of the 9th NECTAR

Conference. Network on European Communication and Transportation Activities Research (NECTAR), Porto, Portugal (CDROM).

23. Kuncheva, L. I., & Whitaker, C. J., 2003. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Machine learning, 51(2), 181-207.
24. Lee, E.H., Kho, S.Y., Kim, D.K. and Cho, S.H., 2020, July. TTP using gated recurrent unit and spatio-temporal algorithm. In Proceedings of the Institution of Civil Engineers-Municipal Engineer (pp. 1-9). Thomas Telford Ltd.
25. Leshem, G., & Ritov, Y., 2007. 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.
26. Li, Y., Fujimoto, R. M., & Hunter, M. P., 2009. Online TTP based on boosting. In 2009 12th International IEEE Conference on Intelligent Transportation Systems (pp. 1-6). IEEE.
27. Lint, H. Reliable TTP for Freeways. Ph.D. Thesis, The Netherlands TRAIL Research School, Delft, The Netherlands, May 2004.Liu, Y., Wang, Y., Yang, X. and Zhang, L., 2017, October. Short-term TTP by deep learning: a comparison of different LSTM-DNN models. In 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC) (pp. 1-8). IEEE.
28. Liu, Y., Wang, Y., Yang, X. and Zhang, L., 2017, October. Short-term TTP by deep learning: a comparison of different LSTM-DNN models. In 2017 IEEE 20th

International Conference on Intelligent Transportation Systems (ITSC) (pp. 1-8). IEEE.

29. Luong, M.T., Pham, H. and Manning, C.D., 2015. Effective approaches to attention-based neural machine translation. arXiv preprint arXiv:1508.04025.
30. Ma, X., Tao, Z., Wang, Y., Yu, H. and Wang, Y., 2015. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. Transportation Research Part C: Emerging Technologies, 54, pp.187-197.
31. Mendes-Moreira, J., Jorge, A. M., de Sousa, J. F., & Soares, C., 2015. Improving the accuracy of long-term TTP using heterogeneous ensembles. Neurocomputing, 150, 428-439.
32. Mnih, V.; Heess, N.; Graves, A.; Kavukcuoglu, K. Recurrent models of visual attention. Adv. Neural Inf. Process. Syst. 2014, 3, 2204–2212.
33. Mori, U., Mendiburu, A., Álvarez, M. and Lozano, J.A., 2015. A review of travel time estimation and forecasting for advanced traveller information systems. Transportmetrica A: Transport Science, 11(2), pp.119-157.
34. Oh, S., Byon, Y. J., Jang, K., & Yeo, H., 2015. Short-term travel-time prediction on highway: a review of the data-driven approach. Transport Reviews, 35(1), 4-32.
35. Oh, S., Byon, Y.J., Jang, K. and Yeo, H., 2018. Short-term travel-time prediction on highway: A review on model-based approach. KSCE Journal of Civil Engineering, 22(1), pp.298-310.

36. Papageorgiou, M., Papamichail, I., Messmer, A., & Wang, Y., 2010. Traffic simulation with METANET. In Fundamentals of traffic simulation (pp. 399-430). Springer, New York, NY.

37. Park, D. and Rilett, L.R., 1999. Forecasting freeway link travel times with a multilayer feedforward neural network. Computer-Aided Civil and Infrastructure Engineering, 14(5), pp.357-367.

38. Ramezani, M., & Geroliminis, N., 2012. On the estimation of arterial route travel time distribution with Markov chains. Transportation Research Part B: Methodological, 46(10), 1576-1590.

39. Ran, X., Shan, Z., Fang, Y. and Lin, C., 2019. An LSTM-based method with attention mechanism for TTP. Sensors, 19(4), p.861.

40. Ruppert, D., Wand, M.P. and Carroll, R.J., 2003. Semiparametric regression (No. 12). Cambridge university press.

41. Schmitt, E. J., & Jula, H., 2007. On the limitations of linear models in predicting travel times. In 2007 IEEE Intelligent Transportation Systems Conference (pp. 830-835). IEEE.

42. Šemanjski, I., 2015. Analysed potential of big data and supervised machine learning techniques in effectively forecasting travel times from fused data. Promet-Traffic&Transportation, 27(6), 515-528.

43. Shen, L., 2008. Freeway travel time estimation and prediction using dynamic neural networks. Florida International University Electronic Theses and Dissertations, Miami, FL, U.S.A.

44. Sun, H., Liu, H. X., Xiao, H., & Ran, B., 2002. Short term traffic forecasting using the local linear regression model.

45. Torday, A., 2010. Simulation-based decision support system for real time traffic management (No. 10-2120). Transportation Research Board 89th annual meeting, Washington DC, United States.

46. Van Lint, J. W., 2006. Reliable real-time framework for short-term freeway TTP. Journal of transportation engineering, 132(12), 921-932.

47. Van Lint, J.W.C., 2004. Reliable TTP for freeways. Netherlands TRAIL Research School.

48. Vlahogianni, E. I., Golias, J. C., & Karlaftis, M. G., 2004. Short‐term traffic forecasting: Overview of objectives and methods. Transport reviews, 24(5), 533-557.

49. Wang, Z., Fu, K. and Ye, J., 2018, July. Learning to estimate the travel time. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 858-866).

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

51. Wunderlich, K. E., Kaufman, D. E., & Smith, R. L., 2000. Link TTP for decentralized route guidance architectures. IEEE Transactions on Intelligent Transportation Systems, 1(1), 4-14.

52. Xu, B. and Qiu, G., 2016, March. Crowd density estimation based on rich features and random projection forest. In 2016 IEEE Winter Conference on Applications of Computer Vision (WACV) (pp. 1-8). IEEE.

53. Yildirimoglu, M., & Geroliminis, N., 2013. Experienced TTP for congested freeways. Transportation Research Part B: Methodological, 53, 45-63.

54. Yu, B., Wang, H., Shan, W., & Yao, B., 2018. Prediction of bus travel time using random forests based on near neighbors. Computer‐Aided Civil and Infrastructure Engineering, 33(4), 333-350.

55. Zhang, S., Zhou, L., Chen, X., Zhang, L., Li, L. and Li, M., 2020. Network‐wide traffic speed forecasting: 3D convolutional neural network with ensemble empirical mode decomposition. Computer‐Aided Civil and Infrastructure Engineering, 35(10), pp.1132-1147.

56. Zhang, Y., & Haghani, A.,2015. A gradient boosting method to improve TTP. Transportation Research Part C: Emerging Technologies, 58, 308-324.

57. Zhao, L. and Chien, S.I.J., 2012. Analysis of weather impact on travel speed and travel time reliability. In CICTP 2012: Multimodal Transportation Systems—Convenient, Safe, Cost-Effective, Efficient (pp. 1145-1155).

58. Zhao, Z., Chen, W., Wu, X., Chen, P.C. and Liu, J., 2017. LSTM network: a deep learning approach for short-term traffic forecast. IET Intelligent Transport Systems, 11(2), pp.68-75.

59. Zhu, T., Kong, X., & Lv, W., 2009. Large-scale TTP for urban arterial roads based on Kalman filter. In 2009 International Conference on Computational Intelligence and Software Engineering (pp. 1-5). IEEE.

60. Zou, N., Wang, J., & Chang, G. L., 2008. A reliable hybrid prediction model for real-time TTP with widely spaced detectors. In 2008 11th International IEEE Conference on Intelligent Transportation Systems (pp. 91-96). IEEE.