DEVELOPMENT OF A BUS ARRIVAL TIME (TO BUS HALT) PREDICTION MODEL USING MACHINE LEARNING

M P V P Medawatte

228014G

Degree of Master of Science

Department of Civil Engineering

University of Moratuwa

Sri Lanka

November 2025

DEVELOPMENT OF A BUS ARRIVAL TIME (TO BUS HALT) PREDICTION MODEL USING MACHINE LEARNING

M P V P Medawatte

228014G

Thesis submitted in partial fulfillment of the requirements for the degree
Master of Science

Department of Civil Engineering

University of Moratuwa

Sri Lanka

November 2025

DECLARATION

I declare that this is my own work, and this thesis does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to the University of Moratuwa the non-exclusive right to reproduce and distribute my thesis, in whole or part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: Date: 05.11.2025

The above candidate has carried out research for the Master's thesis under my supervision. I confirm that the declaration made above by the student is true and correct.

Name and Signature of the supervisors:

Prof. J.M.S.J. Bandara ……………………………. Date: 06. 11. 2025

Dr Loshaka Perera ……………………………. Date: 26/11/2025

Dr Amila Jayasinghe ……………………………. Date: 26/11/2025

Dr Sagara Sumathipala. ……………………………. Date: 27/11/2025

DEDICATION

I dedicate this dissertation to my beloved parents, who have provided me with immense support in everything I do. Your unwavering love, endless support, and boundless encouragement have been the foundation of my journey. I also dedicate this work to my dearest husband, for his immense support, understanding, and love at every step of this academic journey.

M P V P Medawatte

Department of Civil Engineering,

University of Moratuwa.

05.11.2025

ACKNOWLEDGEMENT

I would like to acknowledge many individuals who supported me throughout this study, guiding me and supporting me. Firstly, I am sincerely thankful to the LBS2ITS Project, ERASMUS+ Programme funded by the European Union for partnering with the University of Moratuwa and extending financial support for this research.

I am extremely grateful to the esteemed **Prof. Saman Bandara** for collaborating with me on this study and enriching my research journey immensely.

I would like to express my heartfelt gratitude to my research supervisor **Dr. Loshaka Perera**, Senior Lecturer of the Department of Civil Engineering, University of Moratuwa, who gave me this great opportunity to participate in this study. His invaluable guidance and supervision throughout the entire research study are highly appreciable.

I would also like to thank my research co-supervisor, **Dr. Amila Jayasinghe,** for his provoking discussions that challenged me to think critically and analytically, enhancing the quality of my study. I am sincerely thankful to **Dr. Sagara Sumathipala** of the Department of Computational Mathematics for guiding me through all unfamiliar areas of machine learning and directing my study toward the right path.

I wish to thank the support given by the Department of Civil Engineering, University of Moratuwa and its academic and non-academic staff members. Special mention to **Prof. C. Jayasinghe**, the current Head of Civil Engineering Department in providing all administrative support when required.

I would also like to thank all the lecturers in the Transportation Engineering Group in addition to my supervisor, including **Prof. W.K. Mampearachchi**, **Dr. H.R. Pasindu** and **Dr. G.L.D.I. De Silva,** who helped me both academically and otherwise during my time as a researcher.

I would like to provide a special mention to the postgraduate course coordinator, **Mrs. Melani Jayakody,** for her continuous administrative support and my **fellow research assistants** in the Transportation Engineering Group during the period of 2022/2023 for their constant help and guidance throughout the research period. Finally, I must thank all others whose names are not individually mentioned who had helped me throughout the period of the study.

M P V P Medawatte
Department of Civil Engineering,
University of Moratuwa.
05.11.2025

ABSTRACT

Accurate, real-time bus arrival information is essential for enhancing public transportation efficiency. Although real-time vehicle location data provides valuable information on current bus positions, it alone cannot forecast future arrival times. Prediction models are therefore essential for estimating travel times to upcoming stops by capturing the dynamic relationships among traffic flow, passenger demand, and operational conditions. In Sri Lanka's public transport network, bus arrival times are currently estimated using Google Maps and Google Transit. However, these systems rely solely on static transit data for their predictions, limiting their accuracy in accounting for real-time external conditions. As a result, they do not incorporate dynamic factors that could significantly impact bus travel time predictions. Hence, there is a need to identify unique factors affecting bus arrival times within the Sri Lankan Road system and to develop an accurate travel time prediction model that accounts for these factors. This study focuses on the development of an accurate real-time bus arrival time prediction model using Global Positioning System (GPS) data and machine learning techniques.

GPS data from the Moratuwa to Colombo bus route (100) was collected for the study over 30 days, covering weekdays, weekends, and public holidays, using five GPS units at various times of the day. This included data from 335 bus trips, resulting in GPS data points with an accuracy up to 1 sec. The route consists of 53 bus stops and data filtering was performed based on GPS locations relative to each bus stop to calculate travel times between each stop separately. To improve accuracy, the total route was segmented by considering the number of bus stops, allowing for more precise travel time predictions by identifying speed changes in each section. Through a review of the literature, key factors affecting bus travel time were identified, including road section, time of day, day of the week, peak/off-peak hours, availability of bus lanes, distance travelled, traffic flow, weather conditions, number of signalized intersections, and number of signalized crossings. Data collection was conducted to encompass all these factors.

After data collection, the most accurate travel time prediction model was identified by evaluating various types of models previously used in research. The available machine learning models, the Support Vector Regression Model 9, Adaboost Model, K-Nearest Neighbors (KNN) Model, Random Forest Model, XG Boost Model, Gradient Boosting Regression Model, and Long Short-Term Memory (LSTM) Model were selected to be trained with the available dataset. Mean Absolute Error (MAE), Mean Square Error (MSE), Mean Absolute Percentage Error (MAPE), Root Mean Squared Error (RMSE), and R-squared value were selected as model evaluation parameters. Out of the tested models, SVR, RFR, KNN, AdaBoost, XGBoost and GBR performed similarly to each other, showing minute changes with MAE values varying between 17 seconds to 18 seconds and r^2 values greater than 0.6. It was clear that the LSTM model performed prominently better than the other models due to the time series nature of the data set providing the lowest MAE of 3.8 seconds and the highest r^2 value of 0.98. The proposed model aims to generate accurate bus arrival time predictions for each bus stop when the trip's starting time from the origin is known. The model can be further improved in the future by incorporating real-time traffic data and real-time accident data to enhance prediction accuracy, providing accurate bus arrival times for passengers across the public transport network.

Keywords: Intelligent Transport Systems, Bus Travel Time, Machine Learning, Long Short- Term Memory Model

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

GDP - Gross Domestic Product

ITS - Intelligent Transportation Systems

APTS - Advanced Public Transport Systems

GPS - Global Positioning Data

ML - Machine Learning

LBS – Location-Based Services

ITMS - Intelligent Traffic Management Systems

IPIS - Intelligent Passenger Information Systems

MAE - Mean Absolute Error

MSE - Mean Squared Error

MAPE - Mean Absolute Percentage Error

RMSE - Root Mean Square Error

GIS - Geographical Information System

MM - Map Matching

ATM - Automated Teller Machine

TMS - Traffic Management Systems

ATMS - Advanced Traffic Management Systems

FTMS - Freeway Traffic Management Systems

UTC - Urban Traffic Control

ADAS - Advanced Driver Assistance Systems

AVI - Automatic Vehicle indication

IVIS - In-vehicle Information Systems

ATIS - Advanced Traveler Information Systems

TTI - Traffic and Travel Information

PTP - Public Transport Priority

AVM - Automatic Vehicle Monitoring

AVL - Automatic Vehicle Location

GTFS - Google Transit Feed Specifications

APC - Automatic Passenger Counting

WIM - Weight in Motion

ATCS - Adaptive Traffic Control Systems

ANN - Artificial Neutral Network

AI - Artificial Intelligence

SVR - Support Vector Regression

KNN - K- Nearest Neighbours

RFR - Random Forest Regression

GBR - Gradient Boosting Regression

SVM - Support Vector Machines

RBF - Radial Basis Function

SMA - Simple Moving Average

OLS - Ordinary Least Square

FNN - Fully connected Neural Networks

LSTM - Long Short-Term Memory

API - Application Programming Interface

RNN - Recurrent Neural Network

MdAE - Median Absolute Error

PCA - Principle Component Analysis

CHAPTER 1

INTRODUCTION

1.1 Background

Intelligent Transportation Systems (ITS) have taken a vital role in incorporating next-generation technologies and advancements into transportation [1]. ITS can be identified as the application of Information and communication technology in the field of transportation engineering, along with the required infrastructure and various operating systems. It is the most promising technology that can be used to solve modern-day problems, promoting comfortable and well-structured transport systems [2]. ITS applications will mainly benefit transport network controllers and other users of the system [3].

Traffic control and management have become a major issue for developing countries in the world. Improving existing public transportation facilities and making them more attractive and reliable to use has become one of the most crucial solutions for addressing traffic congestion [4]. Enhancements in public transportation will attract more passengers to public transport modes, reducing the number of passenger cars, which results in improved traffic conditions [5]. Advanced Public Transport Systems (APTS) is an important ITS application that enhances the service quality of public transportation. It is a great tool which develops public transportation services in many ways, making it user-friendly and increasing operational efficiency and ridership [4]. It has been more appealing to passengers since its outcomes greatly contribute to daily commuters of public transportation. It incorporates various information systems to obtain fare, timetable, and passenger information for authorities to assist in making more effective decisions on public transport operations and systems [6].

Real-time information in public transport networks is a key element in improving the level of services provided. Real-time passenger and service information helps to decide the efficiency of the existing transport network [7]. The bus transport network is the most widely used mode of public transport in Sri Lanka, as it extends to a broader area than any other available public transport option. Bus travel time information is a critical factor that directly impacts the reliability perceived by public transit users and plays a vital role in attracting more passengers to public transport. Waiting times for public transport services are also a vital component which deteriorates the satisfaction of public transport users [8] [9]. It is identified as a main cause for the higher level of discomfort in passengers, which makes them uncertain about the arrival times of buses. The use of APTS in providing real-time bus arrival information can decrease the disutility of passenger waiting times and reduce them to a greater extent [10][7][11].

The availability of real-time bus arrival information will permit the passengers to properly plan the trips through public transport, omitting unnecessary delays and longer waiting times and utilize the waiting times more productively [10] [12]. Furthermore, passengers will be able to identify the most convenient and fastest route out of the different available routes and connections according to their needs [7].

Bus travel times are affected by many factors such as traffic congestion, time of the day, weather, road conditions, signalized intersections, road geometry, driver performance and bus dwell times [13] [14] [15]. The predictions can differ according to different combinations of the above factors making the accurate prediction of bus travel times a complex task. Many developed countries in the world utilize bus travel time prediction models incorporating real-time Global Positioning Data (GPS) tracking to provide accurate bus arrival times for bus stops. However, the parameters affecting the travel time are not universal, making it inapplicable to different parts of the world. Furthermore, most of the developing countries have heterogeneous traffic conditions, a lack of lane discipline, and unstable vehicle flow conditions, which have further contributed to the uncertainties in travel time predictions [16], [17], [18].

This study focuses on the development of an accurate real-time bus arrival time prediction model using Global Positioning System (GPS) data and machine learning techniques. While previous studies have employed various techniques such as historical average models, regression, time series techniques, and Kalman filtering for short-term travel time estimations, machine learning techniques have demonstrated higher accuracy levels. This is due to their ability to handle complex relationships among bus arrival time predictors, large volumes of intricate data, and non-linear connections between predictors. It is also considered to function best with heterogeneous, lane-less traffic conditions and with fluctuating traffic volumes, which is similar to Sri Lankan traffic conditions [17], [19], [20]. In comparison with other travel time prediction techniques, ML algorithms have shown the most promising results with 23% better precision [21].

While the General Transit Feed Specification (GTFS) and its real-time extension (GTFS-rt) are widely adopted standards for sharing transit data between agencies and third-party applications such as Google Maps, they do not perform prediction tasks themselves. Instead, GTFS-rt serves as a mechanism for transmitting real-time operational data such as vehicle positions, trip updates, and service alerts to external systems. However, the quality and consistency of GTFS-rt data can vary significantly across agencies, with previous studies reporting frequent feed errors and incomplete updates. Such inconsistencies in real-time data can indirectly affect the performance of machine learning based bus arrival time prediction models that rely on these feeds [22]. Therefore, ensuring the reliability and accuracy of GTFS-rt data is essential to improving the precision of arrival time predictions. This study aims to develop an accurate real-time bus arrival prediction model by leveraging Global Positioning System (GPS) data and machine learning (ML) techniques. Through a thorough literature review, the research identifies key factors influencing bus arrival times at a bus stop. Furthermore, the methodology applied in this research for the selected route can be generalized to other routes, provided the necessary data is available. By integrating real-time bus location data into the model, future enhancements could be done for more predictive accuracy.

In conclusion, the development of an accurate real-time bus arrival prediction model using GPS data and ML techniques has significant potential to revolutionize public transportation systems, especially in heterogeneous and dynamic traffic conditions. By leveraging the advanced predictive capabilities of ML, this approach addresses key challenges such as variability in traffic patterns and the complex interactions of factors influencing bus travel times.

1.2 Objectives of the study

The main objective of the study is to identify the most accurate Machine Learning model out of the available models for bus arrival time predictions at bus halts. This research develops an accurate and reliable bus arrival time prediction model for individual bus halts using real-time GPS data. To achieve this, the study first identifies and analyses the key factors influencing bus travel times based on the literature. Following this, a comprehensive dataset was collected from GPS-enabled public transport vehicles and preprocessed to ensure quality and consistency. Several machine learning algorithms were then developed and tested to model bus travel times along route sections. The best performing model was selected based on its predictive accuracy and generalization capability. Finally, the model's prediction errors were benchmarked against results from similar studies to validate its superior performance and highlight its contribution toward improving public transport reliability and passenger information accuracy.

1.3 Scope of Study

Bus travel time predictions of bus routes in Sri Lanka greatly depend on the nature of the bus route. The factors affecting long-distance bus routes may vary slightly from the travel time predictions for short-distance bus routes to bus stops. This study focuses on the travel time prediction of buses to bus stops for short-distance bus routes. The Moratuwa – Colombo (100) bus route is taken as the study sample route.

1.4 Outcomes

The dissertation includes the following outcomes,

- Identification of factors affecting short-distance bus travel time
- Reviewing previously used Machine Learning models for short-distance bus travel time predictions
- Development of six Machine Learning models for predicting short-distance bus travel times.
- Identification and validation of the most accurate travel time prediction model developed.

1.5 Outline of the Dissertation

The outline of the dissertation is as follows.

Chapter one describes the background of the study, stating the main objectives and the scope of the study, along with the research outcomes.

Chapter two consists of a comprehensive literature review carried out under several sections. The first part of the literature review focuses on Intelligent Transport Systems and how Location Services are incorporated in this field under different categories. It highlights all available ITS applications under Intelligent Traffic Management Systems (ITMS), Intelligent Passenger Information Systems (IPIS) and Intelligent Public Transport Systems. It also highlights the advantages of the above-mentioned ITS applications. The second part of the literature review provides a detailed review of the existing travel time prediction models, including historical average models, regression models, Kalman filtering models and machine learning techniques. The review is then narrowed down to travel time prediction models developed using machine learning models. Different identified machine-learning models are explained in detail, taking the existing research as reference. Then the review discusses the factors affecting bus travel times identified in previous studies. It also focuses on the data collection techniques and performance evaluation techniques followed in similar studies. The review further covers studies carried out in Sri Lanka related to bus travel time predictions.

Chapter Three explains the main framework of the methodology used for the study. It describes the data used in the study and provides a detailed account of the data collection process, introducing the study area of the research. The chapter elaborates on data distribution, data preparation, and data preprocessing. It also includes the feature selection methods used in the study to identify factors affecting bus travel time. The modeling of the selected machine learning models is explained in detail. Model evaluation and validation techniques used in the study are defined, and the process of selecting the best-fitting model is described.

Chapter Four presents the results obtained from the developed models. Each model is evaluated using error metrics including Mean Absolute Error (MAE), Mean Squared

Error (MSE), Mean Absolute Percentage Error (MAPE), and Root Mean Square Error (RMSE). The models are also compared using their R-squared values. Scatter plots illustrating actual travel time versus predicted travel time are included. Finally, the chapter features a comparative analysis that assesses the final results of all the models. It concludes by selecting the best-performing model and discussing potential improvements for applying the model to another bus route, as well as suggestions for narrowing down data collection in such instances.

Chapter Five of the thesis presents the conclusion of the study. It summarizes the research objectives and the methodology employed to achieve these objectives. The chapter also provides a synthesis of the research results. It outlines the challenges encountered during data collection and data analysis, along with recommendations for overcoming these practical difficulties. Furthermore, it offers suggestions for enhancing the developed ML model. The chapter also highlights potential avenues for future research that can build upon the findings of the current study.

CHAPTER 2

LITERATURE REVIEW

The global urban area population is predicted to grow 1.5% per year up till 2030, challenging the countries and cities to maintain the well-being of citizens. It is a vital need to secure a higher standard of mobility in cities for the public [23]. Air pollution, low security, unreliable public transportation, traffic congestion, and insufficient parking facilities are some of the main issues faced within developed cities [24]. Increased traffic congestion within major and minor cities has threatened the day-to-day activities of people in many ways. It has led to a decrease in mobility and accessibility, and increased travel times [10]. Carbon dioxide emissions from vehicles have reached higher levels, leading to excessive air pollution caused by the increased number of vehicles on the road [25]. Out of the various solutions presented for traffic congestion and air pollution, improving public transportation and attracting passengers to public transport modes are considered important. It will assist in reducing the number of passenger cars on the road. In order to enhance the service quality and make public transportation attractive to users, the necessary information and details regarding its services should be presented to passengers[10]. The provision of real-time details associated with public transport vehicles is a key element to elevate the quality of public transport [12].

Providing timely and accurate information at bus stops about bus arrival times and their location to passengers is essential in order to reduce their waiting times. [17]. It will also reduce the queue lines at bus stops, assuring comfortable transitions between different transport modes [26].

2.1 Intelligent Transport Systems

The improvements in ITS have been fast and impressive within the last decade. It conceptually describes the possible avenues to incorporate the next-generation mobile technologies into the transport sector [27]. It has led to numerous improvements in the safety, efficiency, and convenience of the road system, such as reduction of traffic congestion, a decrease in road accidents, better management of traffic flow, reduction of carbon emissions and air pollution, and higher productivity in the usage of resources [28]. ITS is linked with many sectors like urban, highway, railway, water transportation, and aviation fields. Rapidly developing areas such as logistics are using ITS to handle complex information to increase work effectiveness, receptiveness, and transparency [29].

ITS can be put into the most effective usage when it’s linked with its real-time implementations [30]. Most ITS applications require real-time vehicle positioning data through essential components like GPS, Geographical Information System (GIS), and Map Matching (MM) [31]. They use positioning sensors along with a digital map to identify the precise locations of vehicles and their movements. These systems collect all updated information and forecasts regarding traffic and weather conditions in order to be directed to relevant systems. With the development of technology, the use of internet facilities and their coverage, wireless communications, and GIS and mobile positioning systems have led to improved and much simpler Location Based Services (LBS). The basic purpose of LBS is to obtain an accurate location at the right place at the required time. It can be observed that many branches of ITS overlap with LBS that use geographical location referencing in providing various services [32].

With the development of technology, the use of internet facilities and their coverage, wireless communications, and GIS and mobile positioning systems have led to improved and much simpler LBS. The basic purpose of LBS is to obtain an accurate location at the right place at the required time. As location is a critical feature when considering privacy and monitoring, the related applications should be handled in a responsible manner [33]. The geo-processing power of mobile phones has eased the process of obtaining geographic locations of people, as most of the public are mobile users [34]. Therefore, LBS is becoming popular among emerging mobile applications. LBS performs two main actions, which are tracking the location and providing that information to perform a certain service. It can be categorized as triggered LBS services, which automatically provide location information under specific conditions and user-requested LBS, where the user chooses when to obtain location data. Location sensing and awareness are required in many fields at present. The LBS sector has been developing for nearly two decades, focusing on various sectors. It is used in entertainment, tour guidance, advertising and promotions, mobile commerce, navigation, and location searching [35].

2.2 Applications of Location-Based Services in Intelligent Transport Systems

ITS is used worldwide in both public and private transport sectors to improve road safety and monitoring. The development of the transportation field is now directed towards optimizing the available facilities in a well-planned manner through proper maintenance and smooth management. This intention is satisfied by the new avenues opened with the introduction of ITS [36]. Applications of ITS cover many areas in the transportation field. As a broad field, its key tools supporting efficient transportation are involved in almost all sectors related to transport. Many researchers have categorized its applications according to various criteria. As a summary, these applications can be identified under three categories as Intelligent Traffic Management Systems, Intelligent Passenger Information Systems, and Intelligent Public Transport Systems [37].

2.2.1.1 Intelligent Traffic Management Systems (ITMS)

Traffic Management Systems (TMS) play a crucial role in the transportation network. It improves the safety, mobility, flow, and productivity of a system. With the introduction of ITS, it has become a key area where TMS is applied [38]. The main function of an ITMS is to obtain information related to traffic flow, measure and study it in order to arrive at various conclusions. ITMS is operated with various functions working together. These include traffic control, incident management, highway and traffic flow management, demand management, enforcement of traffic regulations, electronic payment systems, and many other areas. They involve operations such as location referencing, data acquisition and processing, communication, and information utilization [39]. ITMS plays a major part in the overall management of transport systems.

Advanced Traffic Management Systems (ATMS) include real-time traffic monitoring and controlling, traffic signal monitoring, variable message signs, speed violation vehicle implementation, and provision of information regarding the road surface status. In traffic monitoring systems used in countries like Turkey, data such as average speed of vehicles, vehicle classification details, vehicle counts, direction, and vehicle speed are obtained by the systems. Bridge and tunnel traffic management systems are also a branch of ITMS that control the entry of vehicles to match the capabilities, reducing unnecessary congestion at tunnels and bridges. Freeway Traffic Management Systems (FTMS) improve the functionality of ITMS systems, providing quick responses to congestion and other incidents. From direct management to dispersal of traffic volumes, it handles larger volumes of traffic and provides information to traffic information centres, ensuring a smooth traffic flow in freeways [40].

Urban Traffic Control (UTC) is a part of ITMS that collects real-time data from cameras and conducts speed analysis to adjust timing in traffic signals and inform road users. The main objective of UTC is to reduce traffic congestion by various means of controlling traffic flow and thereby reducing the possibility of accidents. Timings of traffic signals are adjusted accordingly, even considering the peak hours in road sections. Mainly, sensors attached to roads and signal intersections are utilized in feeding information to these systems. In a study done in Poland to understand the applicability of ITS applications, it was identified that apart from large cities (>1.5M) and medium cities (0.5M – 1.5M), even in small cities (< 0.5M) UTC systems were very effective [2].

LBS applications, such as lane detection systems, are conducted through various autonomous mapping algorithms. These are also used in Advanced Driver Assistance systems (ADAS) as well. The shape and the model of the road are considered when identifying the position of the vehicle with respect to road dimensions. Cameras are used for Road Lane Identification, which involves the processing of images obtained through them. [41]. Video cameras are also utilized to obtain lateral and orientation information respective to road markings. In some applications, these road markings are mobile-mapped accurately by the vehicles themselves. GPS and dead-reckoning sensors are also used in providing location data [42].

Real-time information about vehicle locations is one of the most important data in ITS. This allows the users to navigate or obtain the required information related to their location. Automatic Vehicle Identification (AVI) systems are capable of indicating a vehicle's location at a particular time when it passes through an AVI terminal. By tracking the same vehicle at another AVI terminal along its route, this system can obtain actual travel times along a road network [39]. AVI systems are supported by various infrastructures such as transponders attached to vehicles, video cameras, license plate matching techniques, network-based detection systems, and roadside beacons. Apart from identifying vehicles, it can combine with other ITS systems in proposing travel routes, charging vehicles for road use, identifying stolen vehicles, and monitoring commercial vehicles[43].

Advanced Driver Assistant Systems (ADAS) is another branch of ITS with such integrated technologies. It allows the driver to manoeuvre a vehicle with proper instructions to avoid collisions through expert communications [44]. ADAS assists the primary driving tasks, detects and analyzes the surrounding environment, provides warnings when required, uses signal processing and enhances the vision of the driver [45]. These up-to-date mechanisms incorporate expert sensors and communications to manure vehicles in complex road networks [46].

2.2.1.2 Intelligent Passenger Information Systems (IPIS)

ITS integrate a range of information and communication technologies to enhance transport safety, efficiency, and traveller convenience. Among them, In-Vehicle Information Systems (IVIS) provide functions such as in-vehicle signage and speed limit alerts, improving passenger safety by accounting for driver behaviour, surrounding traffic, and vulnerable road users [47]. Technologies such as Digital Audio Broadcasting and Dedicated Short-Range Communications support these real-time information exchanges [39].

Similarly, Advanced Traveller Information Systems (ATIS) collect and process traffic flow data to identify abnormal conditions and inform road users, as demonstrated in Moscow’s ATIS implementation [40]. Providing predicted travel times is a critical feature of such systems, which rely on Automatic Vehicle Identification (AVI) and related technologies to support informed route selection [39]. Other ITS applications, such as Japan’s Vehicle Information and Communication System and the “probe car system," use vehicle-mounted sensors and navigation data to deliver accurate, real-time traffic information [48].

Traveller information components of ITS, such as on-trip public transport information, route guidance, and navigation, depend heavily on GPS data to determine real-time vehicle locations. Through map-matching (MM) algorithms, this data can accurately identify vehicle positions on road sections and arterials, improving navigation accuracy and driver confidence [49]. Traffic and Travel Information (TTI) systems further enhance these capabilities by providing comprehensive, multimodal data, including traffic conditions, weather, and infrastructure status. European implementations employ protocols such as Datex I, Transport Protocol Expert Group, and Radio Data System Traffic Message Channel for standardized data exchange [46]. For instance, Italian TTI systems integrate inputs from national sources like the National Weather Service, AccuTraffic, and Intellicast Highway Conditions to forecast travel limitations and optimize route planning [50]. These systems also inform travellers about available routes, distances, and public transport options [40].

In addition, ITS technologies support non-motorized transport by supplying information for cyclists and motorists regarding parking availability, restricted roads, and interchange facilities [30]. Advanced systems even interface with smartphone applications to adjust traffic light behaviour and improve safety for vulnerable users [47]. A more recent location-based service (LBS) innovation uses geotagged photos to visualize and analyze tourist movements, integrating spatial and behavioural data through Application Programming Interfaces (APIs) and map-based visualization tools [51]. Overall, these ITS-based systems highlight the growing significance of real-time data acquisition, processing, and dissemination, which form the foundation for advanced predictive applications such as bus travel time and arrival time prediction models developed in this study.

2.2.1.3 Intelligent Public Transport Systems (IPTS)

Intelligent Public Transport Systems (IPTS), a key branch of ITS, have significantly enhanced public transport operations through applications such as intelligent speed adaptation, passenger information systems, transport demand management, fleet management, navigation, electronic ticketing, and public transport priority mechanisms [37]. A major barrier to increased public transport usage has been the lack of accessible and accurate information, which reduces passenger confidence. ITS applications address this gap by providing timely details on routes, schedules, and service frequencies, thereby improving reliability and user satisfaction [52].

Within IPTS, Public Transport Priority (PTP) systems are designed to minimize travel times and improve service efficiency by giving priority to buses at signalized intersections. Techniques such as the Split, Cycle and Offset Optimizer Technique

(SCOOT) dynamically adjust traffic signal cycles to favour public transport vehicles [37]. Similarly, driver assistance systems such as Green Light Optimal Speed Advisory (GLOSA) and Green Light Optimal Dwell Time Advisory (GLODTA) guide drivers on optimal speeds to minimize signal delays [53]. These systems employ vehicle detection and queue management strategies [54], and in some cases, dedicated bus lanes monitored by lane operation systems, which enhance bus service quality and promote modal shifts from private vehicles to public transport [55].

Several cities have adopted IPTS-based applications that improve both passenger experience and operational control. For example, Shanghai's smart bus information system enables real-time vehicle tracking, driver behaviour monitoring, and passenger counting, while displaying next bus arrival times at stops [56]. In Istanbul, intelligent bus stops equipped with LED displays and emergency communication points provide real-time route information via GPS [40]. Advanced IPTS implementations also integrate seat sensors to display real-time seat availability and reservation status, further enhancing passenger comfort and convenience [57].

LBS extend the benefits of real-time data to passengers via mobile applications. The OneBusAway system, developed by the University of Washington, provides real-time bus departure and arrival information, reducing passenger waiting times; its post-evaluation showed 87% user satisfaction. Similarly, the BusView system in the USA monitors over 1000 buses to improve scheduling reliability [58]. Many of these applications operate on smartphones, simplifying access to transit information and increasing user adoption. Google Transit, built on the GTFS, is widely used to provide schedule data and optimize route networks [59]. Research on smartphone-based navigation for public transport identifies mobile trip planning, pedestrian network modelling, and uninterrupted positioning as core requirements for such systems [60]. Another innovation, geotagged smart card data, enables the collection of detailed passenger travel behaviour information, supporting service optimization and network simulations [61].

In Sri Lanka, a low-cost public transport tracking system utilizing GPS and mobile data was developed to enhance commuter information access. The system integrated real-time location data from mobile devices into a central database and provided transit information through a mobile app. Surveys revealed that 42% of commuters previously estimated bus arrival times based solely on experience, while 62% expressed willingness to share location data, highlighting the potential acceptance of such solutions [62].

At the operational level, APTS integrate technologies such as fleet communication, Automatic Passenger Counting (APC), Computer-Aided Dispatching (CAD), and demand management systems to optimize service delivery. These tools improve scheduling, route planning, and service reliability, ensuring efficient use of available resources [63]. APC data in particular supports the evaluation of service adequacy and helps identify areas for operational improvement. Collectively, these ITS and IPTS applications demonstrate the critical role of real-time data collection, processing, and dissemination in improving public transport reliability, efficiency, and user experience. Such technologies form the foundation for predictive models like those developed in this study, which aim to enhance bus arrival time prediction accuracy and strengthen real-time passenger information systems.

2.2.2 Advantages of ITS Applications

The incorporation of ITS has resulted in greater improvements in the safety, efficiency, and eco-friendliness of the transport sector. It has optimized the available transport infrastructure in order to increase the capacities within the transport network. Traffic congestion is inconvenient not only for private vehicles but also for buses, trains, and passengers. Various applications of ITS are used to manage traffic congestion to reduce the journey times of passengers, ensuring customer satisfaction [64]. They allow vehicles to function smoothly during a trip, ensuring comfort and safety. According to Janusova and Cicmancova, ITS can be identified as the most effective solution in ensuring passenger safety, turndown travel times, reducing fuel consumption, and solving many other transportation problems [65]. According to research, the transportation sector is also responsible for 13% - 23% of greenhouse gas emissions [38]. It has also been a reason for ITS becoming popular, as it has significant positive environmental impacts. The two main costs of transportation are due to traffic congestion and accidents. These two areas are properly addressed through ITS applications. Therefore, currently, many countries have implemented ITS in order to increase the standards of the transportation sector and lower the costs associated.

Applications of ITS in the Republic of Korea have increased from 14% to 30% at the end of 2020. Analysis of these applications shows that the average travel speed of vehicles has increased by 15% - 20% and traffic congestion has decreased by 20% with the use of ITS systems [37]. Italy has initiated the use of ITS applications for traffic management since the '80s. Around 50% of the countries' local transport providers use ITS applications in fleet management systems, monitoring the fleets [44].

In a study conducted by Touloki on the advantages of ITS, it was seen through a survey that many ITS users agreed that ITS applications have reduced their travel time, provided them with more environmentally friendly travel options, upgraded the quality of public transport facilities and even increased the income of the population[66]. ITS applications have led to an increase in the traffic flow, which has in turn, reduced harmful emissions and transport pollution even from low fuel-efficient vehicles[64]. ITS applications ensuring uncongested traffic flow are said to reduce 19,000 tons of CO_2 per road section of 1000km, which involves ITS installations. A study conducting a Cost-Benefit Analysis of the sustainability of ITS has identified that ATMS has a cost-benefit ratio of 5.89, considering its initial implementation prices. It was also identified that systems such as AHS and ISA show a negative outcome, as the cost of these ITS-installed vehicles was very much higher[67]. Nevertheless, according to Stevens, CBA should not be the sole scenario in assessing ITS applications, as all criteria that highlight the validity, reliability, and performance of ITS should be taken into consideration [68].

TSP systems have allowed faster cost-effective, and reliable transport services in the implemented countries. In Washington City, the implementation of TSP has reduced the transit signal delay considerably. In Portland, these systems have led to a 19% decrease in the travel time of vehicles. In Chicago, it has shown a 15% travel time reduction while in Los Angeles it showed further improvements with a 25% travel time decrease [1] ,[69]. According to the US Transport Department, TSP systems have reduced the delays in arrival times of vehicles very effectively by 40% percent.

In APTS transit management systems that utilize ITS increase the security and safety associated with the training centre, passengers, and drivers while increasing the efficiency of systems. Transit services become more consistent and reliable with the proper control of the fleet and drivers. The information from these systems can be incorporated into other ITS applications such as traffic flow metering, vehicle passenger counters, and transit information systems [63].

Intelligent Transport security systems have lowered the accidents happening on the road while reducing the threats for passengers [70]. In Russia, these systems have reduced the accidents that caused injuries to passengers by 10%, when comparing accident details from 2014 to 2020. Fatal accidents occurring on the road are reduced by 1.5% [40]. The primary objective of installing ISA systems is to increase road safety by controlling vehicle speed. In a study on ISA systems, it was identified that they have reduced the average vehicle speed by 34% of the vehicles that had exceeded speed limits earlier [71]. Various crash prevention systems and safety systems installed in vehicles have been very useful tools in ensuring safety and accident prevention. According to ITS handbook, these systems are currently used in 75% of trucks and 85% of passenger cars [72].

[73] systems, when installed have resulted in many positive results and have upgraded the public transport services. The application of PTP systems in the Soweto-Parktown corridor in South Africa has given its effects in various aspects. The travel time of public transport vehicles was reduced by 29% and its level of service was increased by 20%. Increasing the private car travel times by 8% also resulted in this [74]. IPTS systems have improved the service quality of public transportation, resulting in congestion reduction and reductions in pollution, leading to lower levels of CO_2 emissions [75].

AMS has shown many improvements in road behaviours dropping red-light violations by 20-70% and improving customer satisfaction by 72%. It has shown a 5-11% decrease in peak travel time while increasing fuel consumption by 2-13% [76]. Through the estimate done by the transportation department in Florida, AMS has decreased vehicle delays on the road by 1.8% and reduced the travel times by 13-48%. A similar application of ITS is HMS. HMS used in the UK shows very positive outcomes when implemented, decreasing the number of accidents by 15%-50%, injuries caused in accidents by 20%-29%, and reducing delays by 46%[73]. The punctuality and reliability of public transport systems are ensured by TMS. With the execution of TMS, studies have shown a significant increase in ridership of 45% and a 9-23% increase in vehicle punctuality. The complaints regarding the transit facilities have also been reduced by 26% [37].

Advanced traveller information systems improve the revenue and ridership of transit while increasing passenger convenience. The use of electronic payments has reduced the bus fare collection time and processing costs and has resulted in more flexible fare structures [63]. Furthermore, the presence of relevant information has resulted in 66-86% of passengers' travel behaviour. It has reduced late or early arrivals by 40%, ensuring the reliability of transport by 5-16% [73].

2.3 Bus Arrival Time Prediction

Bus arrival time information is a very important criterion affecting the reliability of public bus transportation. Therefore, accurate and reliable bus arrival time prediction can entirely change the user perception of public bus transportation, providing accurate information[77]. Significant applications of ITS, such as APTS, utilize automatic vehicle locators and GPS to constantly track bus locations. By installing GPS units in buses, the tracking systems can be continuously operated, providing the arrival times of buses to passengers at bus stops [17]. Apart from that, mobile phones, video cameras, Bluetooth, and radiofrequency devices are used for identifying the geographical location.

ATIS associated with ITS uses advanced Vehicle Location Information to get higher accuracy levels for travel time predictions. These models utilized map-matching techniques that estimate the vehicle location using GIS Software and consider average vehicle speeds [26]. In many studies, AVL data is used to develop travel time prediction models for bus routes along urban corridors. Validating the bus schedule times using the data in order to assess the schedule adherence is done[78].

Bluetooth technology is identified as a successful and low-cost tool to obtain traffic data. Many researchers have studied its applicability in obtaining accurate travel time details in complicated traffic situations. It uses BT MAC address scanners (BMSs) to identify the different Media Access Codes (MAC) in vehicles at different locations, tracking their real-time movement. However, these methods include drawbacks like inaccurate multiple device detections and travel time estimation noises [79].

Also, the importance of short-term travel time prediction is highlighted with the development of ATIS. Small fluctuations in scheduled bus times on links and increased dwell times at stops may cause lengthened waiting times [80].

2.3.1 Short-Term Travel Time Prediction Models

In research on the calculation of travel time, three different schemes were proposed. The first method involves the calculation of bus arrival times depending on bus location and timetable data. The second method involves the prediction of travel times considering the time periods, time of the day, and days of the week. The third method considers the travel times in-between bus stops and the bus dwelling time at the bus stops [13].

There are several different techniques for predicting short-term travel time. These techniques include Historical Average Models, Regression and time series techniques, Kalman Filtering techniques, and Machine learning techniques [17],[10],[81].

2.3.1.1 Historical Average Models

Historical Average Models predict bus travel times based on previously recorded data to give present and ahead travel times. The traffic conditions on the road are considered unchanging. However, these models become reliable when the traffic congestion on the road segment is stable or insignificant [16]. Historical models are comparatively simpler computational models that use average speed and distance to calculate bus travel times [82]. The major downside of this model is that it does not consider the elements that influence travel time into consideration. These methods were used in research that was conducted 5-10 years earlier [83]. Research developing a simple statistical model using historical data for travel time prediction using data of 6 months for a road stretch with 9 bus stops, which has taken the travel time between bus stops into account, showed low MAPE values even for weekday peak hours, implying higher congestion reduces the variability of travel times [16].

In a statistical model developed for predicting travel time by Chung and Shalab, explanatory variables were used, and the inputs mainly consisted of two sections, namely, historical data (previously gathered for a few days) and the present operating conditions. To prevent overestimation of arrival times, certain operation strategies were also included. The model performance was evaluated by comparing it with real-world operation data collected through GPS [26]. When certain models are entirely based on historical data, considerably larger data sets are required to presume traffic characteristics at a specific time of the day [84]. Historical Average Models seem to perform the best when the traffic conditions on a road stretch are consistent daily or weekly with similar patterns at different times of the day [85].

2.3.1.2 Regression Models

Regression models contain a set of independent variables which form a linear function that describes a dependent variable. They are white-box models that come up with mathematical connections between predicted variables and explanatory variables [26], [86].

In contrast to historical models, these models function better in unstable traffic conditions [87], [10]. They estimate the effect of different factors that are independent of one another on the dependent variable. Different inputs for these models can be provided by GPS data connected to buses. Multiple linear regression models have the convenience of identifying which inputs are more dominant or less important for travel time prediction [10],[16]. Linear Regression models are considered accurate when predicting bus arrival times to downstream stops [16]. However, the applicability of these regression models is very low as the variables selected are highly interdependent [80]. These are not advisable to use when traffic data is not available [10]. The applicability of the regression model is narrow, as the changes that can occur to traffic conditions in highly dense traffic situations have many variations [84]. Statistical data is frequently used by regression models. It considers the macroscopic travel times within stops and the microscopic effects on signalized intersections and traffic flow. These traffic-based approaches establish connections between bus spatial-temporal data functions and traffic flow and come up with estimated travel times [14].

2.3.1.3 Kalman Filtering Technique

Kalman Filtering (KF) technique is a model that gathers previous data and forecasts future states based on the conditions in the road stretch [82]. It estimates the current conditions of a system and also becomes the basis of future predictions and improves the predictions by filtering noise[26]. These models give mathematical representations and adapt to traffic variations at different times [80].

A study conducted in Chennai, India, regarding travel time prediction of buses under heterogeneous traffic conditions using GPS data has used algorithms built on the KF technique [17]. This technique allows the representation of traffic variations including time-dependent variables, mathematically [80]. It uses the most recent bus arrival information to be included as input [88].

Studies that had used the KF method to predict travel time and compared it with other methods stated that the results become impractical when the travel times between time steps selected show bigger differences [84]. The KF method conveys the conditions between previous and current time and uses Long Short-Term Memory (LSTM) in obtaining temporary connections [89].

2.3.2 Machine Learning Techniques

With the improvements in ITS, ML has become a very useful tool in resolving complicated problems by extracting patterns from large data sets [90]. Several ML techniques can be used in predicting travel times.

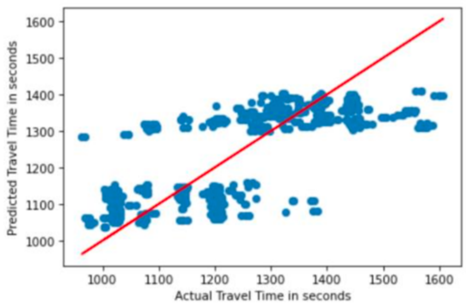
Dynamin Artificial Neural Network (ANN) models can be developed to predict bus arrival times accurately. When doing so, the input-output combinations of data should be selected thoughtfully [10], [17],[91]. ANN is capable of handling complicated relationships among bus arrival time predictors that are caused by large amounts of complex data and non-linear connections between predictors [16],[26],[92],[4]. They try to recreate the brain's ability to identify patterns and experience-based decision-making functions [93]. Different ANN models developed by several researchers have used various sequences of inputs and outputs. Certain variables, such as speed, weather, distance, and flow, are taken as inputs in these models. The use of GPS technology to replace this input data is also possible. Real-time bus arrival and departure times of initial bus stops are taken into consideration through the application of GPS technology to predict bus travel times. In research on developing an Artificial Neural Network bus travel time model using GPS, the previously collected bus arrival and departure time patterns and real-time travel details were both used to predict bus arrival times to upcoming bus stops [10].

Gene Expression Programming is also a black-box model that contains complicated operations. These models need comparatively larger data sets and time-intensive systems training. In a study on bus arrival time modelling in New Zealand, related data for the required routes were obtained for both inbound and outbound directions and combined through SQL and Python languages into a well-organized data set. Through these AVL data, the changes between the real-time arrival time and the scheduled arrival time can be highlighted. By analysing these differences in entrance and exit times at bus stops for consecutive stops, bus travel times were calculated [86].

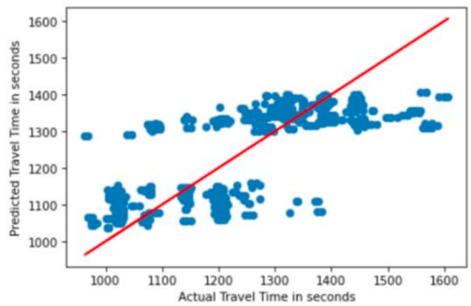
Certain researchers have also studied the application of Artificial Intelligence (AI) to assist real-time traffic management. It is also used in developing microscopic simulation models to assess various traffic management strategies. The study has proposed two AI paradigms, namely, Support Vector Regression (SVR) and Case-Based Reasoning, as decision support tools. These are also capable of identifying real-time traffic diversions due to various incidents[94].

ML models mainly consist of two types: linear and non-linear models. Linear models of ML are faster, simpler and more direct to interpret. They develop linear mathematical equations to estimate unidentified values. If the data you have is not linear, then the efficiency of linear models will be very low. In such cases, non-linear models should be utilized for better results. Non-linear models develop mathematical functions that contain square roots, trigonometric functions, higher degree polynomials, etc[95].

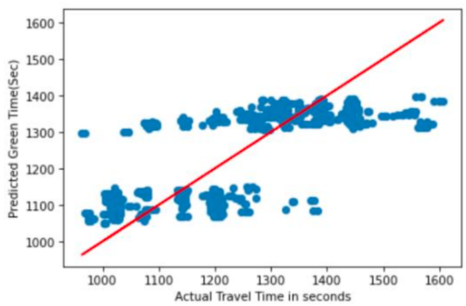
A study in Karnataka, India, on predicting public transit travel time has used eight ML models for comparison. These include four linear models: Ridge Regression, Linear Regression, SVR and Least Absolute Shrinkage and Selection Operator (LASSO) and four non-linear models, such as the K-Nearest Neighbours (KNN) model, Regression Trees, Random Forest Regression (RFR) and Gradient Boosting Regression (GBR). When comparing the predicted values of the above models, the following scattered plots were obtained [95].



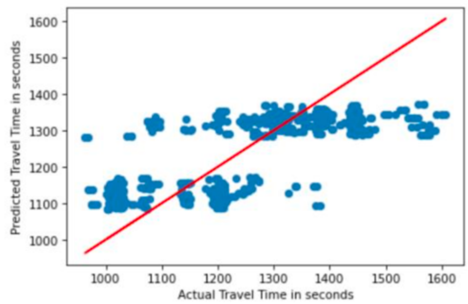
Scatter plot of LR



Scatter plot of RR

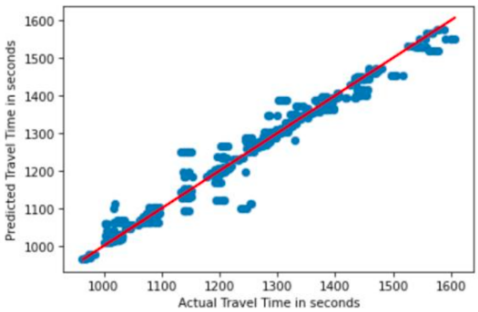


Scatter plot of LASSO

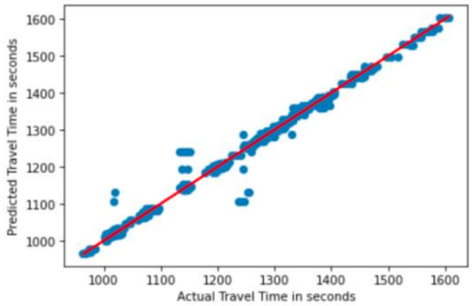


Scatter plot of SVR

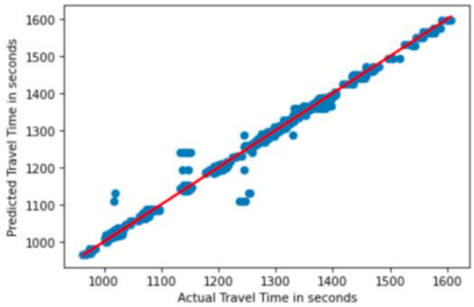
Fig. 2.1: Scattered Plots of Linear Models



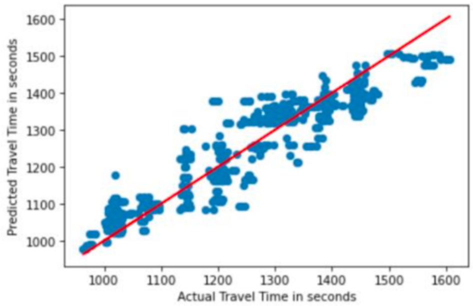
Scatter plot of KNN



Scatter plot of RN



Scatter plot of RFR



Scatter plot of GBRT

Fig. 2.2: Scattered Plots of Nonlinear Models

In Figures 2.1 and 2.2, scatter plot graphs are compared with a 45-degree line, which indicates perfect predicted values. The performance of linear models was very poor compared to non-linear models [95]. Therefore, when selecting machine learning models for the study, more non-linear models were included than linear ones. Below are some of the machine learning models selected to be tested using the dataset in this research.

2.3.2.1 Support Vector Regression (SVR)

The SVR is developed from Support Vector Machines (SVM) for binary response variables. SVR is a regression algorithm, while SVM is a classification algorithm. This algorithm uses residuals smaller in absolute value compared to constants. The performance of SVR surpasses all other regression machines when predicting continuous outcomes[96].

SVR is a collection of controlled learning methods utilized for regressions and classifications [26]. SVR is successfully used in time-series analysis and statistical analysis[97]. Wu et al. have used SVR to predict the travel times in a highway road stretch in Taiwan, depending on the data collected through speed gauges mounted along the road. Here, SVR has shown improved performance over ANN models as they were more flexible to generalization[77]. SVMs are built on the same theory as SVR. It has been one of the most trendy and efficient supervised statistical machine learning algorithms introduced by Vapnik in the 1990s [96]. There are many advantages of using SVM, such as the good generalization capability and the ability to develop even from a small sample size data set [82]. Due to this, it is used in dealing with many issues such as fault card detection, text categorization, speech recognition, etc [96].

A study on SVM conducted in China has analysed the applicability and feasibility of SVM to be used as a bus arrival time prediction tool. This model also compares the historical data with the traffic conditions in real-time. Taking the different traffic flow situations into account is considered important in order to arrive at more accurate predictions. These models require many calculations, and the input variable should be properly selected [93]. SVM models assess them through a test dataset separated from the original data. These depict better prediction abilities with slight average errors in the outcomes [82]. There are several kernel functions, which a linear classifier that resolves non-linear issues, such as the linear kernel function, polynomial kernel function, Radial Basis Function (RBF), and sigmoid kernel function. Many researchers have selected RBF in the predictions based on SVM [98].

A study on bus arrival times to bus stops with multiple routes has used newly developed prediction models based on SVM and ANN applications. It has shown higher accuracy levels with ANN performing better when compared to SVR. Here, the bus schedules are crucial as the bus arrival time from the objective route and every other route should be calculated for the same bus stop [98]. A similar study has used the past data of bus running times through multiple routes to assume the bus arrival time in any of these bus routes. Out of the available four models, SVM, ANN, k nearest neighbours (k-NN) algorithm, and linear regression, it was seen that SVM models functioned better when forecasting bus arrival times [99].

Apart from the above-stated models, several researchers have used hybrid models that have combined two or more models to predict travel times. A study conducted by Liu et al. has suggested various hybrid models built on State Space Neural Networks (SSNN) that required large sets of data and the Kalman filter method. In a study on bus arrival time prediction in China, it was observed that hybrid models performed best among other models, with a Mean Absolute Percentage Error of 17% and with a Root Mean Square Error of less than 1 minute. Gong et al. have used a new approach named Hybrid Moving Average and Dynamic Adjustment (HMADA) to predict bus arrival times. When comparing this model with the already existing model through Mean Absolute Percentage Error, it shows that the model surpassed the accuracy of other historical data-based models [100].

When predicting the travel time in an identified bus route, the total route stretch is divided into several sections, taking the number of bus stops along the route into consideration [17]. Through this, speed changes between two bus stops can be identified, and the changes in the movement of speed can be represented. Segmentation results in more accurate predictions[97]. This segmentation method was also proposed by Ma et al. in research on travel time prediction using the SVM technique. Real-time bus data and taxi data were used to gain the traffic data for the model [101]

The impact of bus velocity at different times of the day was considered in a study, and various models were proposed for travel time predictions. Historical data were used to categorize bus travel times according to different time periods of the day. Four data models were proposed:

- Data models interpreted by bus stops and junctions
- Data models based on the travel speed of buses
- Data models for fixed-distance bus routes
- Data models interpreted only by bus stops [12]

Researchers have identified that SVM methods and ANN methods have provided more accurate and better predictions when compared to other machine learning algorithms like k –NN and linear regression [21] , [20]. SVM algorithms are capable of forecasting several bus arrivals times at a single bus stop[98]. Fully connected ANN (FC – ANN) methods that need a large number of higher-quality data sets surpass the KF models. However, in the availability of limited data sets, SVM performs better than FC-ANN models [20]. The identifiable disadvantage that machine learning algorithms have over the others is that them being harder to share between different systems and are challenging to explain to others [21].

The simplest approach of ML algorithms can be mentioned as Ordinary Least Squares (OLS), which is a component of linear regression. Here, linear combinations are developed on its controlled variables, and the factors depict how each variable contributes to the final output. When detecting the accuracy of these methods, it was seen that the Fully Connected Neural Networks (FNN) method has a higher accuracy of more than 66% compared to OLS for long-term forecasting. Even for short-term predictions, the FNN method showed higher accuracy outputs with 15% more accuracy than OLS. The FNN approach also obtained the lowest Mean Absolute Error when compared with all other proposed techniques [90]. Research on multi-output deep learning methods associated with bus arrival time predictions forwards a multi-model approach that groups three sub-models [89]. The proposed deep learning techniques perform ahead of other OLS and SVR methods [90].

Certain researchers have used automated procedures to identify various types of delays occurring. These involve the collection of data on delays through test drives as well. It will make the system aware of the types of traffic conditions that the vehicles are in. Here, the data required is also less than other methods, making the data collection uncomplicated [102]. Along with geo-spatial data of buses, weather information was also utilized [103].

2.3.2.2 K Nearest Neighbours (KNN) Model

KNN is another non-linear model which is based on the principle of similarity and is commonly known as a lazy learner. It evaluates the variations of the instances in the training test with the test set using a distance formula [104]. Along with other models, many researchers have used this model in comparison with other models in predicting bus travel times. When the KNN model was used for travel time prediction in Guangzhou, it had a 15.99% MAPE value as its performance, which was higher than the neural network model and GBM model [105].

A study conducted in Hefei, China, on bus travel time predictions used 57 million GPS points as historical data to develop a historical pattern model and a KNN model. The KNN model performed well in off-peak hours but was underperforming during peak hours. It was identified that KNN models were not sensitive to sudden changes, such as accident occurrences, due to the models following the historical data [106].

Research conducted under Indian traffic conditions of real-time bus arrival times uses the KNN algorithm as the prediction model. The study has considered a route stretch covering both rural and urban conditions with heterogeneous, lane-less traffic conditions and with fluctuating traffic volumes and different land use and geometric attributes. Due to the presence of heterogeneous traffic conditions with varying vehicle modes, particle filtering is applied to the inputs with the use of k-NN algorithms as a recursive estimation tool. The study uses GPS data collected every 5 seconds from 6 AM to 8 PM for 30 days. When analysing the model performance, the predictions give an accuracy of +/- 2 minutes, also capturing the increased variability of Indian Traffic [19]. The undisciplined nature of traffic makes the prediction models more complex [102].

2.3.2.3 Random Forest (RF) Model

The Random Forest (RF) model is a version of the Decision Tree. It is a non-linear model ideal for handling discrete data in contrast to other models[107]. Random Forest (RF) performs strongly in multi-purpose classifications and regression tasks. It is used to predict continuous variables. In these tree-based models, mathematical equations are not used to identify relationships among different variables. Instead, RF develops models that predict the target variable by studying the decision rules deduced from the feature values of the training data.

A study in India comparing several linear and non-linear models for travel time predictions concluded that Random Forest (RF) Regression performed the best. It showed the lowest error values, with a Mean Absolute Error of 17.99, a Median

Absolute Error of 7.99, and a Root Mean Squared Error of 4.8. The RF model also received the highest R-squared value of 0.9889. Therefore, RF is considered the most compatible model for transit operation planners to upgrade schedules and enhance transit operations. [95].

Researchers have identified the RF approach as the most suitable for forecasting travel times using GPS data [5]. In a similar study, it was concluded that on highway segments gradient boosting method performed better compared to RF in travel time predictions[108],[109]. Research conducted in Beijing using 80,000 bus travel data has utilized RF models for bus travel time prediction. The RF model results showed a prediction error below 4.4% when compared with historical average models. Its prediction error was also 1.24% lower than the Support Vector Regression Machine model. It is identified that when the number of decision trees increased, the final prediction error decreased [110]. Results of another study conducted in Dalian city of China, show that RF models perform even better than ANN models and SVM models in bus travel time predictions. The MAPE value of the RF model was 5.7% compared to the 8.5% MAPE of the ANN model and 8.01% of the SVM model [111].

2.3.2.4 Gradient Boosting Model (GBM)

The Gradient Boosting model is another ensemble model that is comparable to the RF model. It consists of an additive tree model that contains an ML tree model in each level that is utilized to reduce the model prediction errors until it reaches a threshold error value [95]. One of the most important aspects of the boosting method is its ability to resample training data to obtain the most significant data for each model. It is proven that by using boosting algorithms, many weak models can be combined into a single, highly accurate model. In a study conducted in the United States using data from a private sector company, INRIX utilized Gradient Boosting Machine (GBM) for bus travel time prediction across different variables, such as 5-minute ahead, 15-minute ahead, and 30-minute ahead predictions. The GBM model demonstrated superior performance compared to other models, achieving a Mean Absolute Percentage Error (MAPE) of 2.14% for 5-minute ahead predictions, 3.33% for 15-minute ahead predictions, and 3.8% for 30-minute ahead predictions. GBM showed the highest performance for 15-minute and 30-minute predictions when compared with Random Forest (RF) and ARIMA models. This highlights the advantages of using GBM for multi-step-ahead predictions. [108].

The GBM can provide comprehensive outputs using the variable importance. A study conducted in Tianjin, China, predicting bus travel time using a 24-day data set has compared GBM with the Kalman filter and historical mean. The results determined that the GBM model has better MAE and MAPE values compared to the other two, showing that the model has better stability and application prospects when used in travel time predictions [112].

2.3.2.5 Ada Bost Model

Ada Boost, standing for Adapting Boosting, is another ML model that can solve non-linear complex relationships in data. It is one of the most popular boosting techniques that joins weak, erroneous learners, mainly decision trees, with strong, accurate learners.

In a study conducted in Germany Ada Boost model is used for travel time prediction of multimodal freight transport. The model gave a reasonable MAE value, which was 18.78, compared to the Extra Trees model with a 43.66 MAE value. This model was only second to the SVR model, which has a 16.91 MAE value [5].

2.3.2.6 XG Boost Model

XGBoost is an improved learning algorithm derived from Gradient Boosting and Decision Tree algorithms. It is a more efficient execution of the Gradient Boosting model and supports many other weak learners apart from Decision Trees. XGBoost is another model that can be used for more accurate predictions with reduced overfitting. It is used in a wide range of prediction models, such as predicting commodity sales, housing rent, defects in power systems, air quality, prognosis quality score of fracture surgery, and more [89]. It has also been used in prediction models that forecast short-term traffic flow. These applications show how the accuracy of predictions of travel time between bus stations can be further improved using the XG Boost model [113].

In research conducted in Guangzhou using 28-day bus operation data, it was identified that the XG Boost model performed best compared to the Gradient Boosting Model and KNN models, giving the lowest MAPE of 11.96% which was 9.3% below the average of other models. Also, when comparing the model prediction results in different periods at different road sections, it is identified that the accuracy of predictions in non-peak hours is higher, and the accuracies tend to be steadier and more consistent with the increased number of bus stops and road sections [105].

2.3.2.7 Long Short-Term Memory (LSTM) Model

An LSTM neural network comes under the Recurrent Neural Network (RNN), which is a deep learning model that is capable of handling long-term dependencies. It can exclude unrelated information and maintain a cell state by observing previous input data. It is a popular machine learning model in predicting time series, where time series of different window lengths are generated and entered into LSTM models. Time series models depend on data from historical time periods to predict time periods in the future. A different pattern or a pattern that can occur over time is identified, and these are provided by mathematical functions using the given historical data [26].

To study more complicated sequential datasets, multiple LSTM models can be stacked. A study in Denmark has developed a multi-output bus travel time prediction using

LSTM on a dataset of 1.2 M travel time data collected across three months along a bus route using AVL. This LSTM model prediction, when comparing historical average values, has performed significantly better even during peak hours. It has also surpassed Google's Traffic model stemming from live traffic data and a model developed by Movia, a transport authority in the Copenhagen area [20].

A study conducted in Chongquig, China, also uses LSTM model for travel time prediction, specifically due to its ability to remember long-term information and is ideal for problems associated with time series. RNN with LSTM is believed to get precise prediction values at bus stations by correlating several previous stations [114]. Another study conducted in China using bus data from Xingtai bus company for 30 days using the LSTM model to predict bus arrival times, shows the model performs the best for both short-distance and long-distance precision. The result of this research shows an MAE value of 17.11 for short-distance arrival predictions and a 98.07 MAE value for long-distance predictions using the LSTM model [79].

A new ML technique for predicting bus entrance times to bus stops, called Traffic Density Matrix, is used in research. It put up a traffic information depiction method within the selected urban area. To proceed with travel time predictions, various approaches like FNN, RNN, Convolutional Neural Networks, and LSTM approaches are used. The assessment of data is conducted under different outlines and models, such as the Operator Data Model for long-term predictions and the Client Data Model that can be used for short-term predictions. When comparing the model performances LSTM model recorded an MAE value of 56.4, which is only higher than the FNN model, which showed a 44.6 MAE value. The lowest minimum MAE was also reached by the LSTM algorithm with a value of 3.3 sec [90].

A Convolution LSTM model was used for multi-output bus travel time prediction using deep learning for a study in Denmark. The dataset consisted of 3.5 M travel time observations focusing on 32 bus stops. This model performed best for 15 min ahead travel time predictions, giving an RMSE value of 4.38 also showed good performance values for predictions that are 30 min ahead and 40 min ahead [20]. A study conducted using travel time data provided by Highways England uses a 66-series prediction LSTM neural network for 66 links in a bus route. This model had an enhanced performance with a smaller prediction error of 7% median of MREs for the whole 66 links [115].

2.4 Factors Affecting Travel Time

When organizing bus operations, the arrival time of a bus at a bus stop is considered important data to arrive at predictions for the next bus operation. These data can also be utilized for real-time traffic control systems such as signal priority systems. Out of the many factors affecting bus arrival time mentioned above, traffic conditions on the road and signalized intersections show a greater impact. Therefore, incorporating the effect of these factors in models can be seen in research [14].

The main features associated with bus arrival times were identified as the time of departure, bus stop dwell times, real-time travel time, public holidays, average speed, and distance to the destination [11],[78],[105]. A preliminary analysis conducted to understand the correlations of variables in a study in India shows that the time of the day has a bigger impact than the day of the week or the direction of travel [95].

Weekends, school days and holidays also affected the bus travel times. Peak and off-peak hours of a route section are one of the major features considered in bus travel time predictions in most of the previously conducted studies [98],[12],[107].

Apart from these, the traffic flow in a road stretch is decided by traffic lights, road conditions, weather conditions, the number of vehicles on the road, and the nature of driving vehicles [26]. The bus arrival time prediction models can be upgraded if these factors are included as inputs to the prediction scheme [93]. The average speed of vehicles was also used in models for forecasting travel times and durations. Studies in the past have also identified route length, ridership, number of stops and intersections, and geometry of the route as factors affecting bus travel time. [23],[80].

Certain regression models have identified that weather was not a highly effective factor to be included in the model [87]. Ramakrishna et al. discovered that the dwell times of buses near bus stops and intersection delays caused throughout the bus journey from the origin were insignificant as inputs [10].

AVL systems that come with APTS can be used for easier and more precise data collection. The GPS devices in these systems record the speeds of buses, time, and location through points with their longitudes and latitudes. Bus arrival and departure times at bus stops were treated as important data. Functions referring to the time of day, such as the travel time index, were also considered. It was observed that peak hours showed a 20-30% increase in travel times compared to the average travel times recorded. Additionally, it was seen that the travel times of consecutive trip sections or successive buses could be correlated. The total length of the bus route was divided into several sections, and correlation analysis was carried out, noting the travel times through successive sections. [10].

A travel time prediction model, which used GPS data to predict travel time at each bus station, used the input variables as the time of the day, the coded ID of the previous bus station, the coded ID of the current bus station and the travel time taken from one stop to another as input variables [10]. Jeong and Rilett have used schedule adherence as an input variable for their bus arrival time prediction model, which also used arrival time and dwell time at each bus stop. This was to capture the traffic congestion levels by comparing the scheduled arrival time with the actual arrival time at bus stops. They have also focused on the effect of peak and non-peak hours on weekdays and weekends [16].

A study conducted in New Jersey to predict travel time has identified a few variables that determine the bus link travel time, such as average link delay, travel distance of the link, average speed and volume of the link, average queue times and passenger demands at bus stops. Further, it has been identified that the bus stop spacings, number of intersections between two bus stops, and mean and standard deviation of speeds and volumes as variables affecting travel time between two consecutive bus stops [80].

A bus arrival estimation time model in the United States has used data from APC data installed in buses to develop a model. The direction of travel, recorded bus door opening and closing time, date of service, day of the week, trip starting time, number of passengers boarding and alighting were some factors considered in the model. Furthermore, the regression model results showed that variables like weather were not very significant. The unavailability of major variations in weather was considered a reason for this [87]. However, a study conducted in Macau considers weather conditions as an important factor for bus arrival time prediction. The main reason for this is presented as the presence of an increased number of motorcycle users for daily activities by people, which is directly affected by weather conditions [88]. Weather, time period and whether it is a holiday were the main three factors considered in the influence factor design of the bus arrival time model that was developed by the Yuxi intelligent bus system, in China [103]. Another way of considering the effect of weather is considering precipitation as an influencing factor. A study conducted in Beijing considers the traffic data, traffic junction number and also visibility of the road as factors affecting bus arrival time [107].

A study conducted in Shenyang, China, directly used the longitude and latitude data on bus locations along with their current speed and average speed to feed into the travel time prediction model [100]. The nature of the bus route and its urban-rural divide are also considered in research as features affecting bus travel time. It is categorized as a city centre and a suburb [116].

A bus travel time prediction model developed based on Washington DC considers a few detailed variables to predict the total transit travel time of a bus route. Travel time is considered the only dependent variable, while the number of passengers boarding and alighting, passenger load, segment length, bus stops, signalized intersections, access approaches and mid-segment crosswalks are considered independent variables [78].

In research conducted by Chow et al., a kinematic wave model was proposed, and a space-time trajectory of buses was exhibited under normal traffic situations. Calculations were done considering the red-light timings at intersections to determine the delays that occurred [110]. Vehicle queue lengths, queue clearance times, and waiting time for green lights at intersections were also considered in certain research [98].

2.5 Related Studies in Sri Lanka

Research related to bus arrival time prediction models in the Sri Lankan context is relatively limited compared to other countries due to the narrow ITS facilities available within the public transport network. A study conducted by Ratneswaran and Thayasivam [117] on improving the accuracy of bus travel time predictions suggests using XGBoost and Convolutional LSTM models, as these models surpass others by an average of 11% in terms of MAE value. In this study, GPS units were fixed to buses on the 654-bus route from Kandy to Digana, which contains 14 bus stops, for 9 months from 6 am to 7 pm. Data from 36 weeks were used for training, while data from 6 weeks were used for testing. When evaluating the performance of the study, the LSTM model and XGBoost model individually showed MAE values of 43.1 and 41.0 seconds, respectively. The best-performing model was the proposed multimodal model, which achieved an MAE value of 36.2 seconds.

Another study targeting the Kandy–Digana road section used 17.3 million GPS data points collected over three months at 15-second intervals, which were obtained from the GPS database at the Department of Transport and Logistics Management, University of Moratuwa, to predict bus route travel times using the MM algorithm. The bus route was divided into 100-meter segments, as the total route length was 15.8 km. The estimated hourly average travel time values from this method ranged from 36 minutes to 62 minutes [118].

When estimating travel times in Highways of Sri Lanka, several factors such as road condition, land use type, geography and road geometry were considered. Five regression models were developed through stepwise regression from GPS data that was collected over thirteen days along 05 different highways. Out of the five models, the best-performing model showed a 17.6 second standard error with an R-squared value of 70.6% [15].

Research conducted on using dynamic traffic data for travel time predictions discusses the ability to use dynamic traffic data on various links collected from Google Maps distance matrix Application Programming Interface (API). The travel time data collected from manual methods were contrasted with the API travel times, and the variations were identified to be within 5% to 15% range [119].

A study conducted on long-distance bus travel time prediction along the Colombo-Kandy route used GPS data for five months, collected at 10-second intervals. An LSTM model with a moving window strategy was used, and different time steps up to three were developed. Both unidirectional and bidirectional LSTM models were trained, and no noticeable difference was obtained between the two. When considering the model performance, it was identified that the model with one step gave the best performance with a 24.2-second MAE value [120].

2.6 Data Collection

Research conducted on predicting bus travel times has used data collected on various time spans to train and test the model. Table 2.1 below shows a summary of some data collections carried out in the literature referred to.

TABLE 2.1: SUMMARY OF LITERATURE ON SAMPLE SIZES FOR DATA COLLECTION

Papers Referred	Data Type Used	Duration of Data Collection	Case Study Area / Data Characteristics
[10]	GPS data	30 days (Dec 2019)	Bus Route 21G, Tambaram, Chennai. (A discontinuous dataset was used) Continuous data was only collected for 10 days 28 km route length was divided into 100m sections to collect data points
[17]	Previously available GPS data	7 months (Nov 2008 to May 2009)	Bus data of bus line LT11 in Macae, Brazil
[16]	GPS data	6 months (June 2000 to Nov 2000)	Bus Route 60 Houston, Texas. (Data received in 5 sec intervals in a 1.6km stretch with 9 bus stops.)
[108]	Data from private company, INRIX	Year 2012	I-95 Bus route data (Data collected at every 5 minutes of travel time)

[121]	Travel time data from Highways England	Year 2013	M25 highway 66 links (Used Automatic Number Plate Recognition (ANPR) cameras for data extraction)
[87]	APC data	311 bus trips (January 2002 - June 2002)	48 km route stretch in Northeast of United States
[77]	Taiwan Area National Freeway Bureau (TANFB) website data	28 days train data and 7 days test data	Collected in Taipei, Taiwan (Raw traffic information for every 3 min)
[84]	GPS data and APC data	5 weekdays	Bus route no 5, Downtown Toronto Route length of 6.5km
[12]	GPS data	2 weeks	Bus line 1,Tenaska, Dobrava For 25 bus stops
[88]	Data collected from Macao Transport Bureau (DSAT)	1 month	Route 03 and 102 between Macao Peninsula and Taipa Island
[86]	Bus trip and passenger flow data from Bus A's Business Objects tool	1 month (March 2015)	Bus Route 274 and 347, Auckland
[91]	GPS data	Year 2013 (January 01- 31)	Bus Route 78 New Delhi, India
[93]	GPD data	270 trip data within one month	Route No 4 Dalian, China Data collected along 10 time points along the route
[19]	Permanently fixed GPS units	30 days Data collected every 5 sec from 6AM to 8PM	Bus Route 19B Chennai, India
[100]	GPS data	4 days (April 23 to April 26)	Bus route 244 Shenyang, China 11.6km bus route with 20 bus stops
[103]	GPS data	100 trips	15 bus route , Yuxi city, China

		Data collected at 10 sec time intervals	
[122]	GPS data	10 days Data collected at 10 sec time intervals	21-L route Chennai, India 22km long route with 21 bus stops
[123]	Bus trajectory data, and smart IC-card transaction records data	27 days April 24 to May 20, 2018 from 6.00 am to 11.00 pm (3 vacation days and six weekend days)	5 bus routes connected to 1 station in Guangzhou city, China
[89]	Real time AVL system installed in buses	3.5M data points (Jan 2016 – Feb 2018)	4A Bus Line in Copenhagen Focused only on the first 32 bus stops along the line.
[82]	Data recorded through mobile logging device	350 bus trips (07 days) 6 am to 12 am midnight	PJ03 bus route Jaya, 9.4 km route stretch with 23 bus stops
[25]	Real time data from API server	3 months	R3-Y Bus Line, Chiang Mai
[98]	GPS Data	16 working days (oct 2014) Expect holidays and weekends from 5.50 am to 10pm	Bus route 01 Zingong, Sichuan 9.5km route length with 17 bus stops
[78]	Data from Washington Metropolitan Area Transit Authority's AVL system	1 month Data collected only during peak hours (6:00 AM - 9:00 AM & 3:00 PM - 6:00PM)	8 arterial Bus Routes, Washington DC
[20]	Data obtained from real time AVL system	1.2M observations in 06 months (May to Oct 2017)	4A bus line, Movia, Copenhagen (Considers only the first 32 links)
[105]	GPS data	28 days	Central US line Guangzhou (18.3km route section with 25 bus stations)
[95]	GPS data	1 month (Nov 2020)	Route no 205 Tumakuru, India (5.5km route length)

It can be observed that collecting GPS data has been a common approach in previous studies for developing travel time prediction models. Most studies have deemed a data collection period of a month or less sufficient to capture various travel time changes effectively.

2.7 Model Performance Evaluation

The prediction accuracy of models should be checked to evaluate the developed models. The performance of ML models is usually evaluated through performance metrics. Using errors is an important attribute of performance metrics because of its ability to differentiate the final model results. There are different error metrics used in regression models [95]

One error metric used for model performance is MAPE, which considers the difference between previously observed values with the predicted values. MAE and RMSE are some of the common performance measures used in studies. These consider the previously observed arrival times and travel times between bus links and the total count of predicted link times [88]. Another error metric used for model performance is MAPE, which considers the difference between previously observed values with the predicted values [10]. A widely used evaluation metric for regression models, R squared value, a regression score function is also used as a regression score function to analyze the model performance. Figure 2.3 and Figure 2.4 show the trend lines drawn in a study conducted by Ashwini to compare the different models tried out based on RMSE, MAE, MdAE and R-squared values obtained [95].

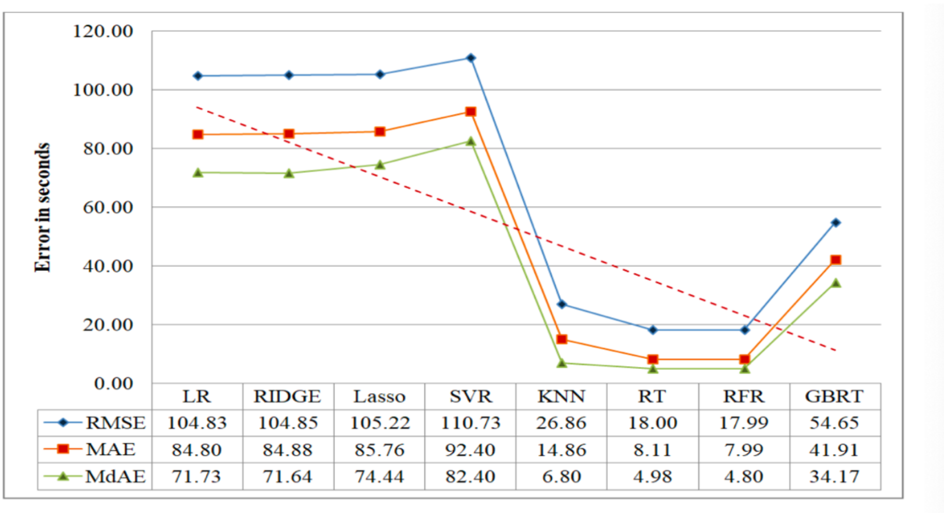


Fig. 2.3: Comparison of RMSE, MAE and MdAE of all models

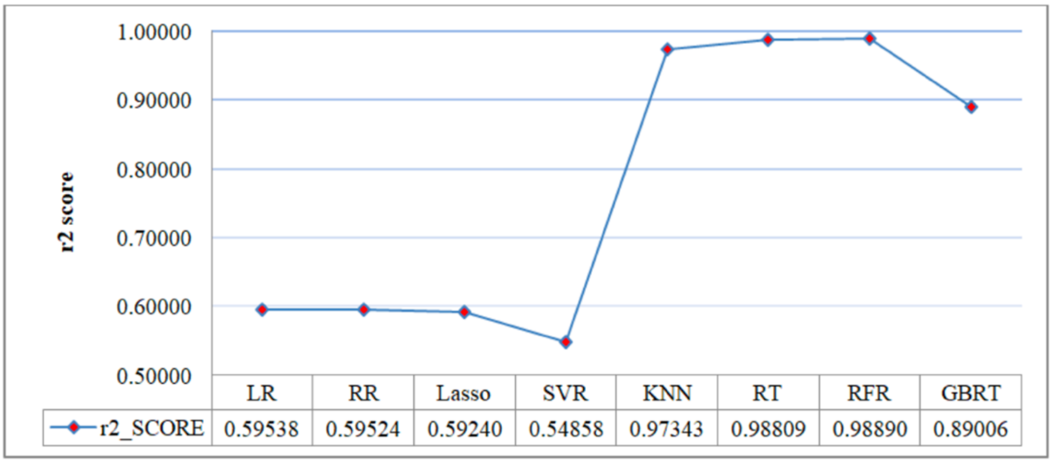


Fig. 2.4: Comparison of the R-squared values of the model

An experimental method known as the ten-fold cross-validation method is used for certain models to depict their appropriateness in predicting traffic conditions[123]. A study done in North Carolina compared the above four travel time prediction techniques, and higher accuracy in prediction values was observed through ANN models. Prediction accuracy of travel times was identified with a maximum of 10% error when the required travel time location is 20 - 50 minutes away from the current location. When this increases by more than 50 minutes, error predictions also increase [10].

2.8 Summary of the chapter

This chapter provides a comprehensive literature review that explores the significant applications of LBS within ITS. It identifies and describes various transportation domains where ITS technologies, based on LBS, are applied, categorizing them primarily into ITMS, IPIS, and IPTS, along with their respective advantages.

The chapter discusses previous research on bus travel time predictions, with a particular emphasis on short-term travel time forecasting techniques. The main travel time prediction methodologies identified include Historical Average Models, Regression Models, Kalman Filtering Techniques, and ML techniques. An in-depth examination of seven ML models is provided: SVR, k-NN, RF, GBM, AdaBoost, XGBoost, and LSTM. The review covers various studies that have employed these models, highlighting their applications and effectiveness.

A descriptive review of factors influencing travel time is also included, identifying variables such as time of departure, bus stop dwell times, real-time travel data, public holidays, average speed, distance to the destination, road traffic conditions, and signalized intersections.

Furthermore, the chapter explores previous studies related to bus travel time prediction, specifically within the Sri Lankan context. It compares sample sizes used in different studies to determine an ideal data collection size for future research.

Finally, the chapter reviews different model performance evaluation techniques utilized in various studies to identify suitable methods for assessing the effectiveness of the models developed in this research to identify an acceptable model evaluation technique for the models developed.

CHAPTER 3

METHODOLOGY

The main framework of the methodology used in this research can be shown as below in Figure 3.1, as a summarized version under four phases:

- Data collection
- Modelling
- Model Evaluation
- Model Selection

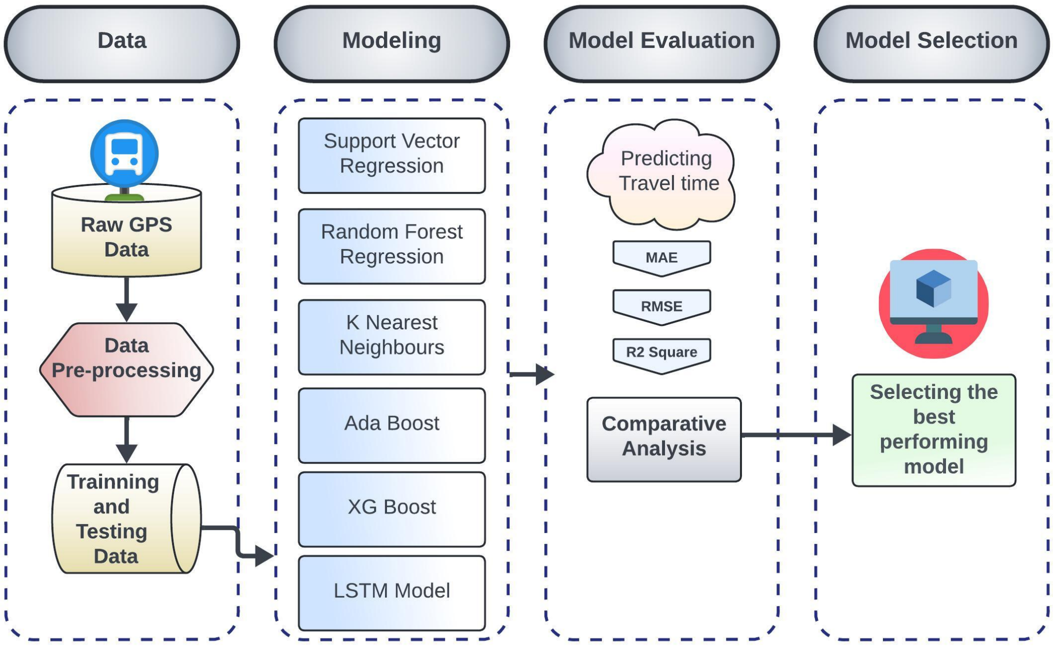


Fig. 3.1:Comparative Analysis Framework

3.1 Data

3.1.1 Data Collection

Bus routes can mainly be considered under two categories long distance and short-distance bus routes. This study focuses on predicting the travel times of short-distance bus routes. Therefore 100 bus route from Moratuwa to Colombo was selected as our study route, which is a 21km route stretch consisting of 53 bus stops (Figure 3.2). Each road segment between two bus stops was divided as a road segment making the whole route stretch divided into 53 road segments. This segmentation was done as this study mainly focuses on the travel time prediction for each bus stop rather than the whole bus route. It also allowed us to identify speed changes between two bus stops and changes in the movement, giving more accurate results [17],[97].

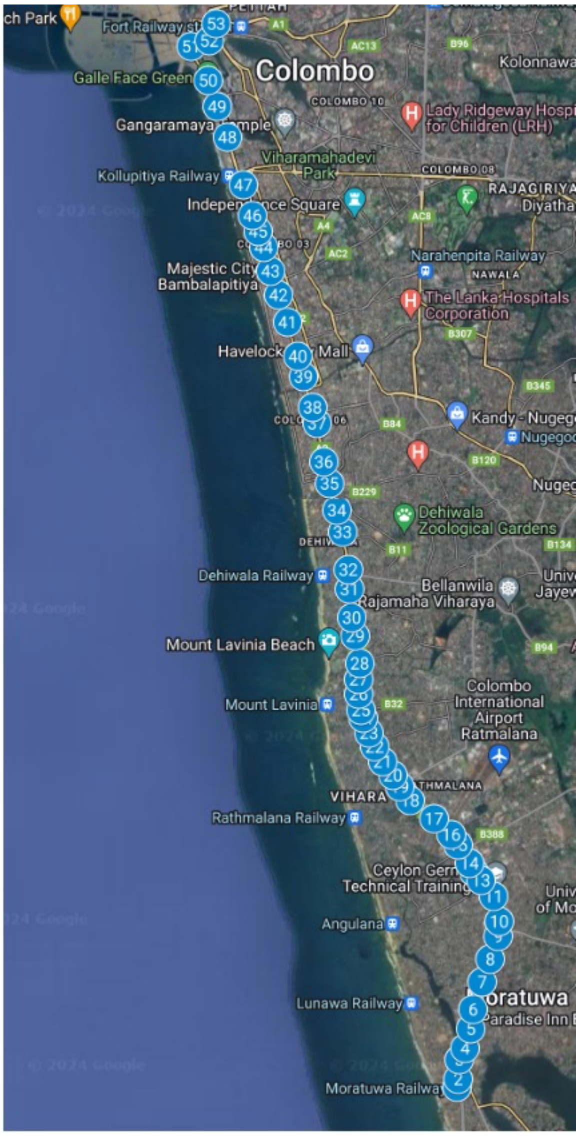


Fig. 3.2: Bus Route Map

Travel time data was collected through five GPS devices for 43 days, covering 335 bus turns, which equated to more than 17,500 GPS data points. The data collection was conducted between March 07th, 2022, to January 10th, 2023. The QSARTS Travel Recorder GPS devices used for the study collected data at 1s time intervals. The data collection was carried out to cover all seven days a week from 6.00 am to 8.00 pm. When referring to Table 2.1 which compares the data sample sizes of different previous research carried out, this data set can be considered adequate. Various erroneous data points were identified. One of the primary causes was the recording of incomplete bus travel time data for an entire journey due to insufficient battery life. Another significant issue was the recording of inaccurate GPS locations, which occurred due to excessive vibrations when buses were stationary, as the devices were positioned at the front of the buses. After filtering the data of incomplete bus turns data of 320 bus turns was used in training the models. Apart from the bus travel time data, data such as route distance were collected through Google Maps. Data such as the availability of signalized intersections and signalized crossings were taken through Google Street View. Weather data was collected manually from the My Weather website, where hourly precipitation information was available [82],[124].

The data collected from the GPS device consisted of 17,800 data points. The data was gathered in a manner that ensured coverage of all possible conditions considered in the factors. Data collection was carried out every day of the week. On weekdays, more than 2,500 data points were collected for each day. Data collected on weekends was slightly lower, with Sundays having the fewest data points recorded. The data collected was limited from 6 am to 8 pm because of the bus travel time schedules. Very limited data was collected before 6 am. The number of data collected on peak hours was relatively higher than in other hours, considering the increased bus turns during that time.

The weather data was divided into four criteria according to the precipitation rate per hour. The precipitation rates for different weather conditions are as follows: On a sunny day, there is no precipitation, with a rainfall rate of 0 mm per hour. Light rain is characterized by a rainfall rate of less than 2.5 mm per hour. Moderate rain falls at a rate between 2.8 mm and 7.6 mm per hour. Heavy rain occurs when the precipitation rate exceeds 7.6 mm per hour, indicating a significant downpour. The distribution of the collected data according to weather is as follows.

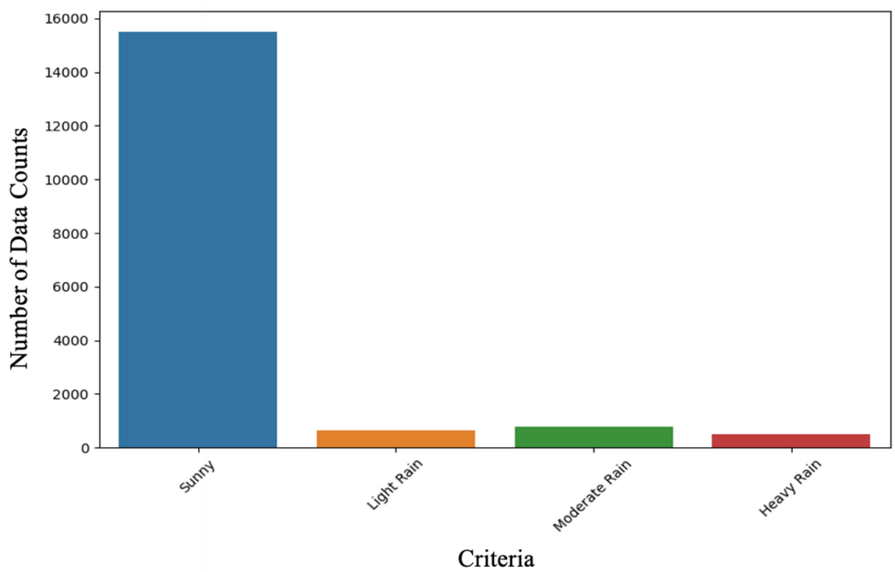


Fig. 3.3:Data Distribution Considering Weather in the Data Set

As shown in Figure 3.3 most data were collected in sunny weather, and a significantly smaller amount of data was collected on the other three criteria depending on the days the data was collected.

In this study, a clear distinction is made between bus travel time, the time taken for a bus to traverse between two consecutive stops and bus arrival time, which refers to the expected time of the bus reaching a specific stop. While the predictive model developed focuses on estimating bus arrival times at stops, the collected dataset primarily consisted of travel times between stops, including variations in bus dwell times at each stop. Incorporating dwell times within the travel time measurements ensures that the model captures realistic operational conditions, including passenger boarding and alighting delays. Both measures are equally important: accurate travel times between stops form the foundational input for deriving bus arrival times, while precise arrival time predictions directly impact passenger information systems and real-time transit management. By integrating both aspects, the model not only predicts the temporal progression of buses along the route but also reflects variations at individual stops, providing a comprehensive and operationally relevant tool for improving public transport reliability.

3.1.2 Data Preparation

The data collected through the GPS devices recorded the local time and date, UTC time and date, longitude, latitude and height data and speed and distance data. Figure 3.4 shows how the recorded bus route GPS points were plotted.

The GPS devices given to different buses were kept on for the whole day for continuous data collection. Therefore, it was required to separate the data of different turns, removing the stationed data in the middle. Also, it was required to identify the time point at which a bus arrived at each bus stop throughout the bus turn.

Initially, the bus stop location coordinates were collected from the data collected through the GTFS project. This data was overlapped with the obtained bus travel data points. Although bus stop locations were obtained, it was observed through the data set that the buses were not stopping at the exact bus stop location and were stopping at locations which are before or further away from the bus stop. Therefore, in order to separate the data points of bus stop arrival times, the whole data was filtered considering their coordinates. Any data point which was within ±10 m radius of the coordinates of the bus stop was taken into consideration as shown in Figure 3.5. After filtering those, the exact time point where the bus comes to a stop was identified and marked as the arrival time to the bus stop by considering the first point of the stationary points.

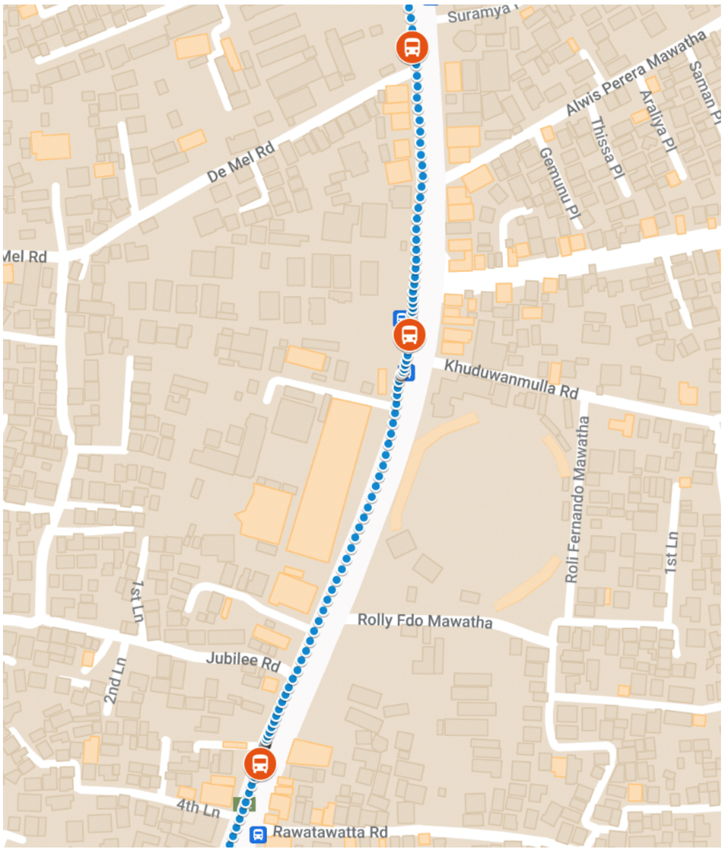


Fig. 3.4:Recorded Bus Route GPS Points in 1sec Intervals

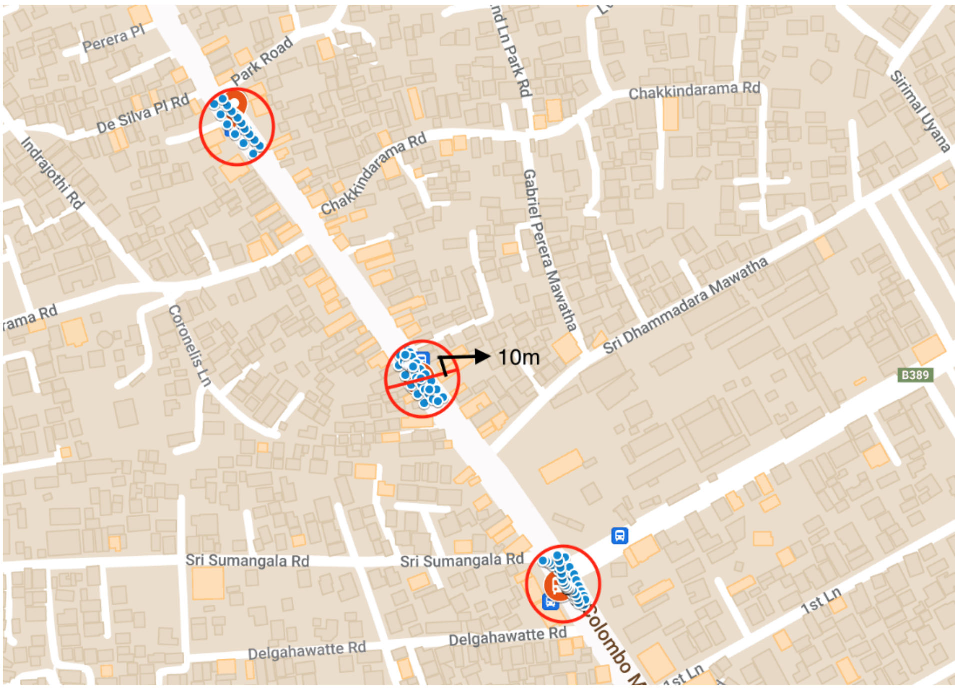


Fig. 3.5:Filtered Data at Bus Stops

After identifying the arrival times, a data table was updated considering the arrival times to all bus stops and the data of 335 turns were logged into the data set.

3.1.3 Data Preprocessing

The raw data logged by the GPS devices needed to be pre-processed before applying it to models, since the data was noisy and erroneous. The following steps were carried out to reduce the errors and irregularities in the data set.

Step 1: All these incomplete bus turn data were removed from the data set, and only the data sets which have continuously recorded data from the first bus stop to the last two bus stops of the route, numbered as bus stop 01 to bus stop 52 or 53, were considered in the study.

Step 2: The Interquartile range method was used on the filtered data, which used quartiles to remove outliers from the data set to preserve the overall distribution of data. Initially, the travel time data were sorted according to each road segment and then filtered using the Interquartile Range method. It uses the first Quartile (Q1), which is a value to which 25% of the data is below and the third Quartile (Q3), which is a value to which 75% of the data is below. The difference between Q1 and Q3 is known as the Interquartile Range. The interquartile range multiplier was taken as 1.5, which is the rule of thumb in determining outliers. Under this standard method, all the data values that are below Q1 - 1.5 * IQR or above Q3 + 1.5 * IQR are considered as outliers. Through this, we were able to filter travel time values which exceeded all other values for each segment due to various external reasons.

The above filtering methods are important to remove outliers and improve model performance. It also improves the data generalization, avoiding overfitting and preventing misleading conclusions. After these filtering methods, the data set was reduced from 17,464 data points to 16,692 data points, giving data of 321 bus turns. As the data obtained was enough to train models data sets with any missing values included were removed from the data sets. Therefore, no missing value identification or fixing of the missing values was done.

Data encoding techniques are used in machine learning models in order to interpret categorical variables which contain different labels or categories into a numerical format. Different encoding techniques can be used in models depending on the nature of the variables, features of the data set and the specifications of the data set. These include one-hot encoding, Label encoding and feature hashing for a larger number of variables. In this research, the Label encoding technique, which is a straightforward, widely used encoding method that is compatible with many machine learning models, was used in this study.

Data scaling is an important aspect in a prediction model for improved model performance. In this study, mainly two data scaling techniques are used: as Minmax Scaler and the Standard Scaler. Both data scaling techniques are used, and the results are compared to obtain the most effective scaling technique for each model. The Minmax scaling technique scales and squeezes the data in each variable into a specific range of values between 0 and 1, while the Standard scaling removes the mean and scales the data set to a unit variance, in a way that it has 0 mean and a standard deviation of 1.

3.1.4 Feature Selection

In order to select the factors affecting bus travel time, a thorough review of the literature was conducted as stated in Section 2.1 of Chapter Two. The factors selected are as below.

- Availability of Signalized Intersections
- Availability of Signalized Crossings
- Hour of the Day
- Distance
- Day of the Week
- Weather Conditions
- Availability of Bus Lane
- Traffic Flow

The variables Availability of Signalized Intersections and Availability of Signalized Crossings were used to identify delays in each route stretch caused by signal cycle times at intersections or crossings. Non-signalized intersections or crossings were not taken into account because the delays they caused were irregular and could be included in the total travel time. Time durations could be categorized in two ways: Peak/Non-Peak Hours or Hours of the Day. The Hour of the Day variable was selected because it provided a narrower window for identifying travel time variations, leading to more accurate travel time estimates.

Additionally, data related to the day of data collection was considered in two ways: Weekdays/Weekends or Each Day of the Week. However, grouping data into weekdays and weekends reduced the accuracy of travel time predictions. Therefore, data was analyzed for each Day of the Week to better capture travel time patterns specific to each weekday.

The variable “Availability of Bus Lane” was selected as the route consisted of two sections, one with a bus lane and one without a bus lane. In various sections of the route, a marked bus lane was available, and in other sections, no marked bus lane was available apart from the two-lane road section. Therefore, through this variable, the effect of bus lanes on bus travel time variation was captured as shown in Figure 3.6 and Figure 3.7.

Fig. 3.6:Route Section with Bus Lane

Fig. 3.7:Route Section without Bus Lane

Data under each feature was collected. Data related to the availability of signalized intersections, signalized crossings, total lanes and bus lanes were collected through visual observations along the route. Data such as distance, bus travel time were collected through GPS devices, which recorded time and location data at 1s intervals. Traffic flow data was given as a categorical variable under four categories, considering the peak and off-peak times along the route.

Feature analysis was conducted to identify the importance of each feature in the model. By doing this feature selection, we could separate the most effective features and select them for the model instead of making it more complex. The feature selection was conducted using the correlation coefficient. The correlation coefficient identifies the linear relationships between the selected variables and the target variable. However, non-linear relationships cannot be identified using correlation coefficients only.

Principal Component Analysis (PCA) is a dimensionality reduction method that transforms the data into a separate coordinate system. It decreases the dimensions in the data while retaining as the variations in the data as possible. It grades the different features relative to their contribution to the data variance, and features will be assigned certain weights according to that. Features with less involvement in the variance will have lower weights. PCA helps in noise reduction, improves model interpretability and improves the training efficiency [125].

3.2 Modeling

Machine Learning (ML) models are algorithms that gain insights into patterns and relationships from data to make predictions or decisions without being programmed to do so. By referring to the literature, this research chooses six ML models for bus travel time predictions and compares them to identify the best-performing model.

Python is used as the programming language when developing the models. It is a high-level interpreted dynamic programming language which is simple and readable. It is a popular and easier option for prediction models because it has abundant documentation and resources, flexibility, simple and readable syntax and cross-platform capability. It also has excellent library support, such as sci-kit-learn, TensorFlow, PyTorch, and Keras, which are high-level APIs for developing prediction models with minimal coding. It also incorporates data science tools and libraries such as pandas for data manipulation, NumPy for numerical computations, sci-kit learn for machine learning and matplotlib and seaborn for 2D plotting and other data visualizations [126].

This study uses Jupyter Notebook as the Integrated Development Environment (IDE) for implementing the Python code [95]. Research carried out on bus travel time predictions has used various simple and complex machine learning models. Some used various deep-learning models and time series as well. By referring to literature in 2.2.4

and 2.2, this study selects the below six models to develop bus travel time prediction models, to compare the results and to select the best model.

- Support Vector Regression
- Random Forest Regression
- Gradient Boosting
- KNN Regression
- Ada Boost
- XG Boost
- LSTM

3.2.1 Parameter Optimization

Parameter Optimization, also known as hyperparameter tuning, is an important aspect in the development of machine learning models. There are different parameter optimization methods available with different advantages and disadvantages. Some of them are grid search, random search, Bayesian optimization, gradient-based optimization, tree-based optimization and meta-learning. Out of the available optimization techniques, grid search, which is a clear and straightforward method, was used in this study when developing the model. It allows us to provide it with a defined set of hyperparameters and tries out all possible combinations of these hyperparameter values.

3.2.2 Development of Predictive Models

3.2.2.1 Support Vector Regression (SVR)

Numerous travel time prediction models in previous research have used SVR models, resulting in significantly strong performances. Therefore, an SVR model was developed with the obtained data set to compare its performance with other models.

Kernel: The kernel parameter defines the type of kernel function used when converting input features into a dimensional space and affects the flexibility and performance of the model. Table 3.1 shows different kernel functions available.

TABLE 3.1: KERNEL FUNCTIONS OF SVR

Kernel Function	Description
Linear	This kernel feature is used when the number of features is larger and when the features have an almost linear relationship with the target variable.

Polynomial	This kernel converts the features unto higher dimensional polynomial functions.
Radial Basis Function (rbf)	This kernel captures nonlinear complex connections between the features and the target variable.
Sigmoid	This kernel converts the features by utilizing a hyperbolic tangent function when the relationship between the target variable and features are nonlinear and has a sigmoidal shape.

In the model developed the kernel function is selected as "rbf" and the kernel coefficient is selected as the default value, "gamma". It is calculated through the equation below.

Equation 3.1: Kernel Coefficient Equation

$$\text{Kernel Coefficient} = \frac{1}{\mathrm{n}_{\text{features}}\ x\ X.var}$$

$n_{features}$ = Number of features

X.var = Variance of the training data

Regularization Parameter: This parameter plays an important role in maximizing the margin of the decision boundary and minimizing the training error. The parameter "C" is given a default value of 1.0. A higher C value narrows the margin, minimizing the training error, but can result in overfitting of the models. A smaller C value is used in simple models, which can generalize more to unseen data.

Epsilon (ε): A SVR model intends to fit a regression line having a specific margin of tolerance outside the real target values. The epsilon parameter specifies the size of this tube. It determines the margin of tolerance for the predicted values. The default value of epsilon is 0.1.

Cache size: Defines the size of the kernel cache, which determines the memory size which stores immediate calculations. The cache size used in the model is the default value of 200.

Tolerance (tol): This parameter tells the stopping criteria of the optimization process and tells when to stop further iterations, as it will not improve the model. The tolerance value used in the model is the default value of 0.001.

The SVR model was developed by changing the above-mentioned parameters until the best performance values were obtained. Both scaling techniques, Minmax scaler and Standard scaler, were used to select the scaling technique that better accommodated the model. PCA was used as a preprocessing technique to reduce the dimensionality of the features.

3.2.2.2 KNN Regression

The KNN regression model, which is a simple and impactful machine learning algorithm for regression, was also used in many previous studies when predicting travel time. Therefore, this model was developed from the available data to test its accuracy in a similar environment. Different parameters are used in KNN models to increase the model's fitting to the dataset.

n_neighbors (K): This parameter decides the number of nearest neighbors that are considered when making a prediction. When a smaller value is selected as 'K' value it allows more flexible decisions but also makes the model more sensitive to noise.

weights: KNN uses various weightage methods to provide more weight to nearer neighbors. There are two weighting methods available as uniform and distance. Uniform weighting allows all neighbors to collaborate equally on a prediction, while distance weighting decides the weightage as the opposite of their distance, giving a higher weightage to closer neighbors.

algorithm: Algorithm parameter refers to different algorithms used in the model to determine the nearest neighbors.

metric: This parameter gives a distance to be considered between data points and the nearest neighbours. The 'euclidean' distance metric is used when continuous data is present. It is equal to the straight-line distance between the two points. 'manhattan' distance is the total absolute difference between the two coordinates of the points. 'minkowski' distance is controlled using the 'p' parameter and is a combination of both manhattan (when p=1) and euclidean (when p=2) distances.

In order to identify the optimal value for the number of nearest neighbours, a range of k values was provided. Since the choice of k value has a significant impact on the model, having a range of k values instead of a single value allows for monitoring the model performance at different complexity levels.

Parameter Optimization: In the developed model, the Elbow method was used to determine the optimal value for k. After assigning a range of k values, the model was trained for each k value. The performance of the models was then evaluated using the

MSE. These performances were plotted on a line graph with the respective k values. The Elbow point on the graph was selected, which is identified as the point where the model performance begins to level off and no longer shows significant improvements when the k value is further increased. The k value at the Elbow point is considered the optimal k value. Apart from the k value, all parameters mentioned above were adjusted to obtain the best-performing model under the given data set.

3.2.2.3 Random Forest Regression

When referring to the literature, RFR models were identified as a popular travel time prediction model. Therefore, this model was selected in this study to observe its performance with the given data set. RFR consists of several parameters which can be changed to control the model's performance and behaviour.

n_estimators: This parameter describes the number of decision trees in the model. The model is more likely to perform better with increased features. However, it can also escalate computational costs.

max_depth: This parameter gives the maximum depth of each decision tree. When the depth is increased, the model can better perform in complex relationships. Nevertheless, it can also lead to model overfitting.

max_features: This parameter decides the number of features to include for the best split. The model has the ability to obtain diverse trees when the number of features increases. However, this can also lead to overfitting.

random_state; This parameter acts as a control for random number generation. By setting a random state, it can be ensured that the same results will be obtained from the model if you run it with the same random state. It ensures that the randomization is repeatable.

bootstrap: When this parameter is set to True when bootstrapping, each tree in the model will be trained on a bootstrap data sample, which will prevent overfitting and ensure randomness.

Criterion: This parameter defines the function that measures the quality of a split in each tree. For regression, these criteria can be MAE or MSE.

min_samples_split: This defines the minimum number of samples needed to split an internal node. When higher numbers are set as minimum sample split it can reduce overfitting but in return, it can cause underfittings.

min_samples_leaf: This parameter defines the minimum number of samples that should be included in a leaf node.

max_leaf_nodes: Defines the maximum number of leaf nodes that a decision tree in the forest can have.

max_samples: This parameter defines the number of samples to obtain from the dataset to train the individual decision trees. If it is set to none each tree in the model will be trained with the entire data set if the bootstrap is set to True.

The RFR model is developed by changing the above parameters and selecting the most fitting features to the data set. Different scaling methods are also used. The model was also tested with and without preprocessing techniques like PCA.

3.2.2.4 Gradient Boosting Regression Model

Gradient Boosting Regression models are identified as another effective travel time prediction model, which can handle complex data relationships. They are also capable of handling large amounts of data. This model is also developed from the available data set to see its performance against the other models. There are different parameters used in this model.

n_estimators: This parameter, referred to as the number of trees, decides the number of boosting stages to be conducted. If the number of selected trees is high, it also makes the model complex and can increase the chances of overfitting.

learning_rate: This parameter proportionates the involvement of each tree. If more trees are present, lower values can be selected to generalize the models.

max_depth: This parameter controls the depth of each decision tree by setting a maximum depth.

min_samples_split, min_samples_leaf: These parameters decide the structure of separate decision trees.

subsample: This parameter gives a fraction of samples that can be used in the fitting of base learners. This can avoid overfitting.

3.2.2.5 Ada Boost

The Ada Boost model is a technique that joins different weak learners to generate a strong classifier. Although the Ada Boost model is used in a few studies for travel time predictions, it has not been thoroughly studied for its ability to predict bus travel times in different conditions. Therefore, this study is expected to trial this model and compare it with other models to test its performance against the others. The Ada Boost model too has several parameters to optimize its productivity.

n_estimators: This parameter decides the number of weak learners to be merged. The performance of the model can be enhanced when the number of estimators increases. However, it will raise the computational costs as well.

learning_rate: This parameter manages the involvement of each weak learner in the final combination. Having smaller values as the learning rate can narrow each weak learner's contribution.

base_estimator: This parameter gives the weak learner which is used as the base estimator for boosting. The default base estimator is 'deprecated'.

random_state: This random state parameter handles the randomness in the training data set and ensures the replicability of the results.

3.2.2.6 XG Boost

XG Boost is another model which is based on the concept of the Gradient Boosting model. Previous studies which have used Gradient Boosting and XG Boost models for travel time predictions claim that XG Boost models performed better than Gradient Boosting models. Therefore, this model is also attempted in this study to examine its adaptation to the given data set. This model also contains a wide range of parameters.

booster: Different types of boosters can be used for this model such as, 'gbtree' for tree-based models, 'gblinear' for linear models and 'dart' for dropout-based models.

objective: The objective parameter in XG Boost assigns the learning task and informs the model of the type of problem that must be solved.

3.2.2.7 LSTM

The LSTM model is another ML model that has been widely used in previous bus travel time prediction research. It is quite an appropriate model to be utilized when previous travel time data is available. A time series is a data type that tracks the changes over time. LSTM is a type of RNN and is very effective for time series prediction work, like bus travel time prediction[126]. A time series data set includes a time column in its structure. However, it is not considered a variable. Therefore, to run the LSTM model in the data set, a time stamp was included containing the date, month, year, hours, minutes and seconds. After uploading the prepared data set dictionary is used to encode all the categorical variables in the data set, which is a straightforward method. It assigns a numerical identifier to each unique element in the data set. When considering a data set of one bus turn, it included the bus stop arrival times of 53 bus stops. However, in some datasets, a few data points were missing in the latter bus stops. Therefore, data sets containing at least 50,51 or even 52 data points under bus stops were taken into consideration. Then, the 52 travel time data points of each bus turn are considered as one data frame, and the whole dataset is split into different data frames.

Similarly, for the LSTM model, a feature matrix of size 52 by 8 (No of features) is passed as input. Initially, two LSTM layers are selected. A few parameters are selected below in this model.

Activation: This parameter controls the availability of information in a cell. Linear, Sigmoid, Tanh and ReLU (Rectified Linear Unit) are available. Since functions such as sigmoid and tanh control the information flow, the activation function used in the model is Relu to improve the learning.

To improve the model performance and minimize the loss function, an optimizer algorithm is used in the training process. The Adam optimizer, which adapts the learning rate of every parameter, is used here. The number of epochs is an important factor in deep learning models. It is the number of which the whole data set is passed forward and backward through the learning model. In order to identify the number of epochs, the validation loss should be monitored. Initially, when the model starts to learn from the given data set, the loss is decreased.

However, if too many epochs are assigned to the model, it can cause overfitting, and the loss can start to increase. The epoch numbers should be tracked to identify the point at which validation loss starts to increase after decreasing. Scaling techniques, Minmax Scaler and Standard Scaler, are both used in modelling to identify the most suitable scaling technique.

3.3 Model Evaluation and Selection

Test train split is a fundamental approach that is used in machine learning for model evaluation and validation. It separates the available data into two sets a training set and a testing set. The model is developed using the training data set. A testing data set is used to evaluate the models by making predictions and comparing them with the actual values. The ratio of splitting of data is decided upon the number of data available. Data split can be done as 50-50, 70-30, 80-20 and 90-10 [127]. This study implements this validation technique in order to find the accuracy of model predictions. The data is split in 80-20 ratio to be taken as training data and testing data.

The evaluations of the above-prepared models were done mainly using the error metric, MAE, as it gave the direct variation of the travel time in seconds. MAPE and R-squared values were also considered as measures of effectiveness in this study. RMSE was also computed. Scattered plots were used for graphical representations. The scatter plots for actual travel time in seconds vs predicted travel time in seconds were plotted based on the results obtained from the test train split.

The selection of the best-performing model was done by performing a comparative analysis by comparing the values under the above-mentioned metrics, MAE, RMSE and R-squared values. Out of these, MAE was mainly considered in the study as it directly stated the deviation of travel time predictions in seconds. The models were also analyzed based on R-squared values. The model with the R-squared value closest to 1 and the least MAE was selected as the most accurate model for predicting short-distance bus travel time.

3.4 Summary of the Chapter

This chapter provides a comprehensive overview of the research methodology employed in the study. The methodology is organized into four key sections: Data Collection, Modelling, Model Evaluation, and Model Selection. Initially, the chapter introduces the study area and details the methods used for data collection, encompassing both primary and secondary data sources. It elaborates on the various data distributions based on the selected features and includes a thorough explanation of the GPS data filtering process. Subsequently, the chapter describes the feature selection techniques employed in the research, presenting the results obtained from these methods. It also covers the data encoding and scaling techniques utilized to prepare the data for analysis. The core elements of modelling ML algorithms are then discussed, including the specific parameters for all seven models. The chapter outlines the parameter optimization techniques applied to enhance model performance. Finally, the chapter specifies the model evaluation techniques used to assess the performance of the developed models, leading to the selection of the most effective model.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Feature Selection

As discussed in the methodology, feature selection was carried out in two as: correlation coefficient and F score. Figure 4.1 is a heat map showing correlation coefficients among the selected variables. Correlation coefficient values measure how closely two variables are related to each other. A Correlation coefficient value of +1 implies a perfect positive correlation where the variables have a linear relationship with each other, while -1 is a perfect negative correlation. Values of the correlation coefficient varying from 0.7 to 1 show a strong positive correlation, while values from 0.7 to 0.3 show a moderate positive correlation.

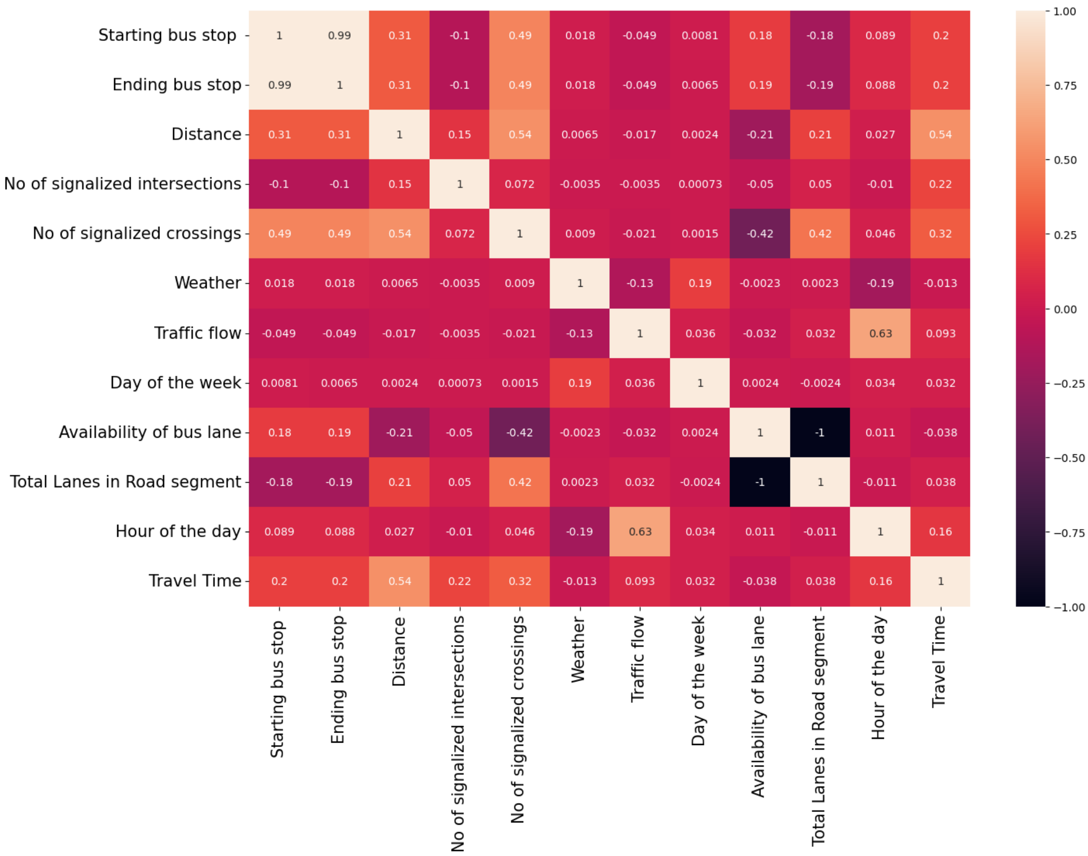


Fig. 4.1: Correlation coefficient of Variables

When considering the correlation among the selected variables, it was identified that variables "Availability of bus lane" and "Total lanes in the road segment" were in a perfect negative correlation with each other. This was because, in this road segment, an additional lane was added whenever a bus lane was introduced in certain segments. Due to this, the variable "Total Lanes in the Road segment" was dropped, and only the "Availability of bus lane" variable was selected. Also, the variables "Traffic flow" and "Hour of the day" showed a high positive correlation value of 0.63. Therefore, the variable traffic flow was removed, and the hour of the day variable was selected. Also, variables Distance and Travel Time showed a moderate positive correlation with a value of 0.54. Variables such as Hour of the Day and Day of the Week exhibited very weak positive correlations, indicating a nonlinear relationship between them. Similarly, the variables Travel Time and Hour of the Day also showed weak positive correlations. Several trials were conducted in this research to reduce the amount of data collection required for future studies. One such approach was broadening the Day of the Week variable to Weekdays and Weekends to simplify data collection. However, this resulted in a reduction in accuracy by 5 seconds per route section. Similarly, the Hour of the Day variable was broadened to Peak and Off-Peak Hours, which further reduced accuracy by approximately 10 seconds. As a result, moving forward, none of the variables were broadened, and they were used in their original form to maintain accuracy.

Under the F score method, when the variations between groups are larger than the variations within the group, a larger F value is obtained. It indicates that the mean values are different in at least two of the groups. This states that the categorical variable shows a considerable effect on the continuous variable that is evaluated. The significant F value was considered as 0.05. When f values are greater than this critical value, it can be concluded that there is a significant variation in group means. The f values obtained from the variables selected in this study are shown in Figure 4.2.

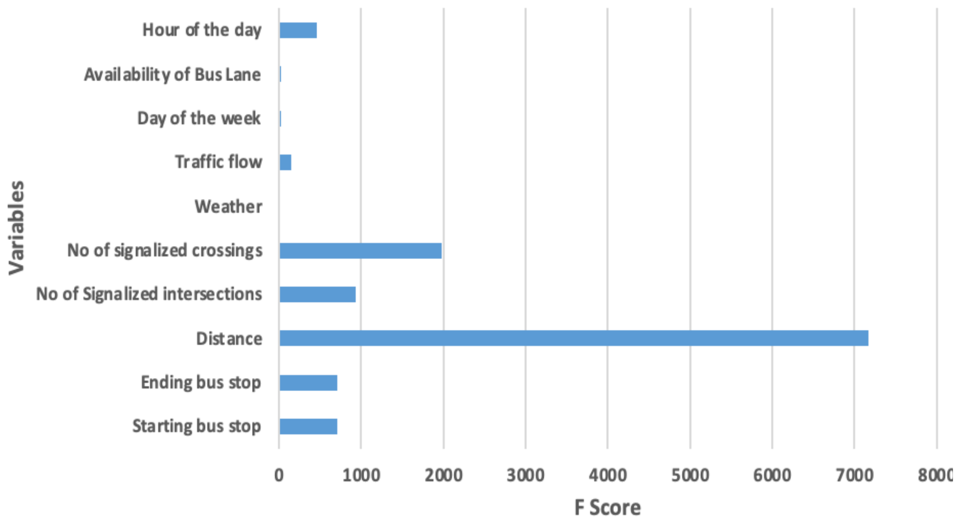


Fig. 4.2: F Score Graph

Through the F scores obtained, we could observe that the variable distance shows the strongest relationship with the dependent variable, travel time. Factors such as the number of signalized crossings, the number of signalized intersections, the route segment and the hour of the day also show higher f-score values compared to others, implying a stronger effect of those variables on travel time.

The final selected variables for bus travel time prediction in this study were as follows.

- Starting Bus Stop
- Ending Bus Stop
- No of Signalized Intersections
- No of Signalized Crossings
- Hour of the Day
- Distance
- Day of the Week
- Weather Conditions
- Availability of Bus Lane

The feature characteristics of the selected variables for bus travel time prediction models in this study are shown in Table 4.1.

TABLE 4.1: FEATURE CHARACTERISTICS

Variable	Data type	Value	Sample Size	Percentage of data (%)
Starting and Ending Bus Stop	Categorical	0-52	335 turns for each bus stop	100.0
Number of Signalized Intersections	Numerical	0	13,098	75.0
		1	4,363	25.0
Number of Signalized Crossings	Numerical	0	8,443	48.35
		1	5,012	28.70
		2	2,328	13.33
		3	1,340	7.67
		4	338	1.94
Hour of the Day	Numerical	5AM - 7AM	4,444	25.48
		8AM - 10AM	4,701	26.96
		11AM - 1PM	4,079	23.39
		2PM - 4PM	3,430	19.67
		5PM - 7PM	785	4.50

Distance	Categorical	< 100m	1	1.96
		100m - 300m	12	23.53
		300m-500m	28	54.90
		500m-700m	9	17.65
		700m-900m	1	1.96
Day of the Week	Categorical	Monday	2,970	17.03
		Tuesday	2,936	16.83
		Wednesday	2,595	14.88
		Thursday	2,787	15.98
		Friday	2,689	15.42
		Saturday	2,057	11.79
		Sunday	1,408	8.07
Weather Conditions	Categorical	Sunny	15,565	89.14
		Light Rain	774	4.43
		Moderate Rain	640	3.67
		Heavy Rain	482	2.76
Availability of Bus Lane	Categorical	BL	11,650	65.96
		NBL	6,011	34.04
Average Speed	Categorical	0-10KM/H	1	1.9
		10-20KM/H	20	38.5
		20-30KM/H	29	55.8
		30-40KM/H	2	3.8

The data can be categorized into two types: numerical and categorical. Categorical data is qualitative, while numerical data is quantitative. For the numerical variables, the mean, standard deviation, minimum, and maximum values of the dataset were analyzed. Feature importance values, assigned by the machine learning model, indicate the relative significance of each variable in making predictions. It was observed that variables such as the number of signalized intersections and crossings, as well as the starting and ending bus stops, had higher importance values compared to other variables.

4.2 Model Training and Evaluation

4.2.1 Support Vector Regression

Grid Search was used to finalize the parameters as explained in the methodology. Table 4.2 shows the parameters associated with the model and the selected parameter according to the grid search.

TABLE 4.2: PARAMETER SELECTION OF SVR

Parameter	Selected Parameter
kernel	rbf
C	1500
epsilon	1.8
gamma	auto
cache_size	100
tol	0.001

However, to reduce the model's complex nature, only the parameters that were most effective for the results were retained. Therefore, parameters 'C' and epsilon were used in the final model. The model was run using both Standard and Minmax scaling techniques. Table 4.3 shows the results of model evaluation parameters obtained from both scaling techniques.

TABLE 4.3: MODEL PERFORMANCE EVALUATION OF SVR

Scaling Technique	MAE	MSE	RMSE	MAPE	R Square
Standard Scaler	17.717	669.587	25.876	0.249	0.618
Minmax Scaler	19.498	868.291	29.467	0.266	0.505

From the results obtained above, it can be concluded that the scaling technique Standard Scaler shows slightly better results than when the Minmax Scaler is used.

4.2.2 Random Forest Regression

The parameters for the above model were also selected through Grid Search, giving the final selected parameters as below. Table 4.4 shows the parameters associated with the RFR model and the selected parameter according to grid search.

TABLE 4.4: PARAMETER SELECTION OF RFR

Parameter	Selected Parameter
bootstrap	True
n_estimators	1000
criterion	squared_error
max_depth	10
max_features	sqrt
min_samples_leaf	2
min_samples_split	10
n_jobs	-1
verbose	2
random_state	53

When filtering the parameters for the most impactful parameters on the model performance, parameters max_features, min_samples_split, random_state and maximum_depth were retained in the final model. This model was also run using both scaling techniques. As shown in Table 4.5, the results show no significant difference in using the Standard scalar and Minmax scaling techniques. Both scaling techniques performed almost the same when comparing the evaluation parameters. The scaling technique was selected as the best-fitting technique for the above model.

TABLE 4.5: MODEL EVALUATION OF RFR

Scaling Technique	MAE	MSE	RMSE	MAPE	R Square
Standard Scaler	17.284	611.381	24.726	0.253	0.651
Minmax Scaler	17.283	611.299	24.724	0.253	0.651

4.2.3 KNN Regression

Table 4.6 shows the parameters associated with the KNN Regression model and the selected parameter according to grid search.

TABLE 4.6: PARAMETER SELECTION OF KNN REGRESSION

Parameter	Selected Parameter
algorithm	ball_tree
Leaf-size	30
metric	manhattan
metric_params	None
n_jobs	-1
n_neighbours	11
weights	uniform

The parameters in the KNN Regression model were selected using Grid Search as well. Out of the above parameters, only the parameters that made a significant difference in the performance were used to reduce the running time and complexity of the final model. The parameters n_neighbours, algorithm, and metric were specified in the final model. The two scaling techniques, Standard Scaler and Minmax Scaler, were separately used in the final model to find the most appropriate scaling technique. Table 4.7 shows the results of model evaluation parameters obtained from both scaling techniques.

TABLE 4.7: MODEL EVALUATION OF KNN REGRESSION

Scaling Technique	**MAE**	**MSE**	**RMSE**	**MAPE**	**R Square**
Standard Scaler	18.145	673.327	25.949	0.263	0.616
Minmax Scaler	18.216	675.887	25.998	0.265	0.614

Through the results obtained above it can be seen that the standard scaling technique performed slightly better in the final model with the above data set.

4.2.4 Ada Boost

The Ada Boost model parameters were identified as below in Table 4.8.

TABLE 4.8: PARAMETER SELECTION OF ADABOOST

Parameter	**Selected Parameter**
base_estimator	DecisionTreeRegressor
max_depth	5
learning_rate	0.01
loss	exponential
n_estimators	200
random_state	100

It was observed that all the parameters considered above had a significant impact on the results. Therefore, all parameters were specified in the final model. The results of the model when the scaling techniques are used are stated in Table 4.9.

TABLE 4.9: MODEL EVALUATION OF ADABOOST

Scaling Technique	**MAE**	**MSE**	**RMSE**	**MAPE**	**R Square**
Standard Scaler	17.480	627.969	25.059	0.255	0.641
Minmax Scaler	17.458	627.056	25.041	0.255	0.642

From the above results, it can be observed that the Minmax scaling technique provided slightly more accurate results than the standard scaling technique.

4.2.5 XG Boost

When developing the XG Boost Model, the selected parameters are shown below in Table 4.10.

TABLE 4.10: PARAMETER SELECTION OF XG BOOST

Parameter	Selected Parameter
booster	gbtree
objective	reg:squarederror
mearning_rate	0.1
Max_depth	3
min_child_weight	1
subsample	0.5
Colsample_bytree	0.5

However, to make the model less complicated and reduce the model running time, only the most effective parameters were selected. Therefore, the booster parameter was retained in the model. Table 4.11 shows the results of model evaluation parameters obtained from scaling techniques, Minmax Scaler and Standard Scaler.

TABLE 4.11: MODEL EVALUATION OF XG BOOST

Scaling Technique	MAE	MSE	RMSE	MAPE	R Square
Standard Scaler	17.446	630.786	25.115	0.252	0.640
Minmax 17Scaler	17.446	630.786	25.115	0.252	0.640

As seen from the model evaluation parameters, the accuracy values obtained by using both scaling techniques were the same.

4.2.6 Gradient Boosting Model

The parameters defined in the Gradient Boosting model are shown in Table 4.12.

TABLE 4.12: PARAMETER SELECTION OF GBM

Parameter	Selected Parameter
learning_rate	0.2
n_estimators	250
max_depth	3
min_samples_leaf	1

min_samples_split	10
loss	'squared error'
random_state	500
subsample	1.0

Out of the above-selected parameters, the parameters that showed a significant influence on the results were selected in the final model, namely, n_estimators, learning_rate and min_samples_split. When designing the above model using the selected two scaling techniques, the following results are shown in Table 4.13.

TABLE 4.13: MODEL EVALUATION OF GBM

Scaling Technique	MAE	MSE	RMSE	MAPE	R Square
Standard Scaler	17.397	618.265	24.865	0.253	0.647
Minmax Scaler	17.402	618.869	24.877	0.253	0.647

It could be seen that the Standard Scaling technique fitted marginally better than the Standard scaling technique in the above model.

4.2.7 LSTM

For data sets with continuous travel time data, including timestamps, LSTM has been frequently used in previous research. LSTM is a model which is specially designed to handle sequential data, which makes it more comfortable to deal with time series analysis. The model is also capable of identifying long-term dependencies. LSTM fits perfectly for travel time predictions, which often use multiple variables for predictions, as it manages multivariate time series data well. The model developed in this study is a multivariate time series model. Initially, the format of the data set is adjusted according to the LSTM requirements. The timestamp of the data was prepared to have the date-time format. The prepared data set was loaded and encoded using the Dictionary encoding technique. When considering the data set, the data included in a single bus turn included 52 bus stops. However, for some data sets, all 52 bus stops were not included due to various inaccuracies. Therefore, bus turn data, which included 50 or more bus stops, was included in the model, withdrawing data sets having below 50 bus stops, which was a negligible count. The data was split into lists and stored as data frames to separate each bus turn, having data of 50. Afterwards, all the lists are loaded, and the features associated with each bus stop are calculated as a loop and stored in an array. Afterwards, features in both input and output arrays were scaled using both the minmax scalar and the standard scaler.

'Pickle' module is used in Python to save a trained model to reuse it with new data sets when making predictions, without the need to retrain the whole model again. The scalar is therefore saved as a pickle file to be reused when making predictions with new data. The LSTM model is input with a feature matrix with a shape of 52 x 7 and is passed through two dense hidden LSTM layers. The 'relu' activation code is used in the model for hidden layers as it is more computationally efficient, and to avoid the weights getting squashed and the learning getting limited. The model is compiled taking 'mse' as the measure of the error/loss, and that error is optimized using an evaluation matrix taken as 'adam' in the model.

In developing the LSTM model for bus arrival time prediction, the dataset was divided into training and testing subsets using an 80:20 split ratio. Accordingly, 80% of the data was used for model training, allowing the LSTM to learn temporal patterns and dependencies within the bus travel time sequences, while the remaining 20% was reserved for testing to evaluate the model's generalization capability. This method ensures that the model's performance is assessed on unseen data, reducing the risk of overfitting and providing a more reliable indication of predictive accuracy. The division was conducted after data preprocessing to maintain temporal consistency, ensuring that the chronological order of trips was preserved so that the model was trained on earlier observations and tested on later ones. This approach reflects realistic operational conditions where future bus arrival times are predicted based on previously observed patterns.

To decide the number of epochs, initially the MAE and MSE are plotted against the number of epochs as shown in Figure 4.3 to identify the appropriate epochs number The above graph shows that the scaled loss values do not go further below a certain point and are maintained at the same level. Therefore, the number of epochs is selected as 2,852. The parameters of the final LSTM model can be summarized in Table 4.14.

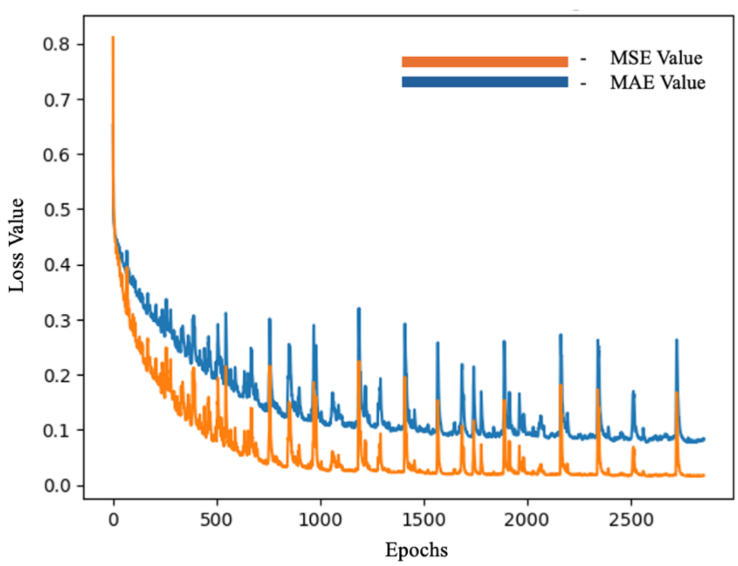


Fig. 4.3: Graph of Number of Loss Values VS Epochs (Scaled)

TABLE 4.14: PARAMETER SELECTION OF LSTM

Parameter	Selected Parameter
No of hidden layers	2
No of hidden units per layer	128
activation	relu
optimizer	adam
metrics	mae
loss	mse
epochs	3000
batch_size	16
verbose	-1

To ensure that the developed LSTM bus arrival time prediction model did not overfit, several diagnostic and preventive techniques were applied. The number of training epochs was carefully selected based on these observations to stop training before the validation loss began to increase. In addition, the 80:20 train test split ensured that model evaluation was performed on unseen data. Regularization techniques such as Dropout layers, early stopping, and batch normalization were considered to improve the model's generalization capability.

The model evaluation parameters of the finalized LSTM when using both scaling techniques are as below in Table 4.15.

TABLE 4.15: MODEL EVALUATION OF LSTM

Scaling Technique	MAE	MSE	RMSE	MAPE	R Square
Standard Scaler	3.826	36.028	6.002	0.032	0.983
Minmax Scaler	4.692	51.380	7.168	0.035	0.976

It was clearly seen that the Standard scaling technique complemented better with the LSTM model giving more accurate results. The coding related to the LSTM model are attached in Annex A.

4.3 Comparative Analysis

The predicted travel times through each model were plotted in a scatter plot against their respective actual travel times in order to identify their differences and recognize the errors of the models. It was compared with a graph that was plotted for the ideal scenario where all actual travel time values were the same as the predicted values, which was a 45-degree line [95] as shown in Figures 4.4 to 4.10.

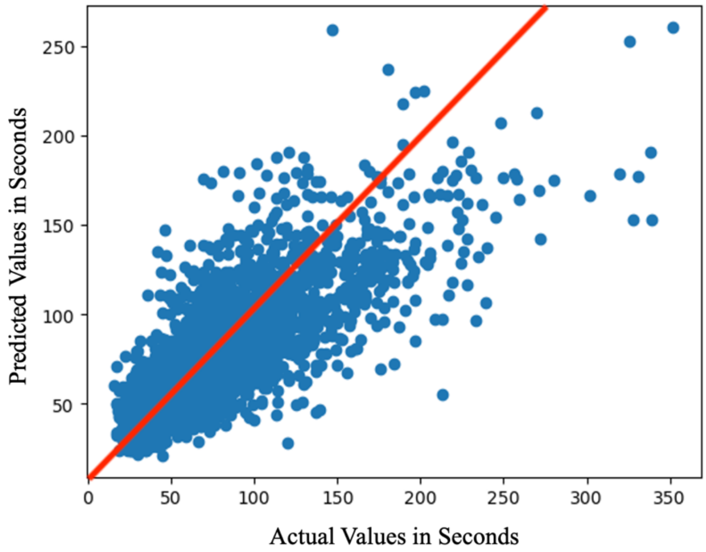


Fig. 4.4: Scatter Plot of SVR

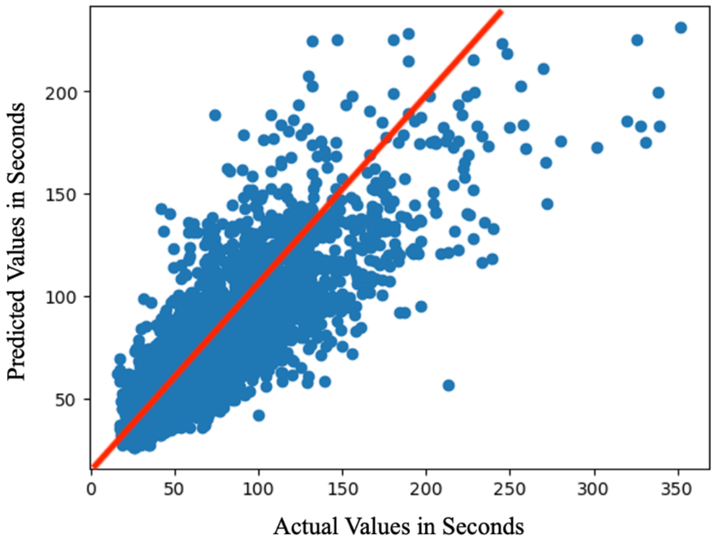


Fig. 4.5: Scatter Plot of RFR

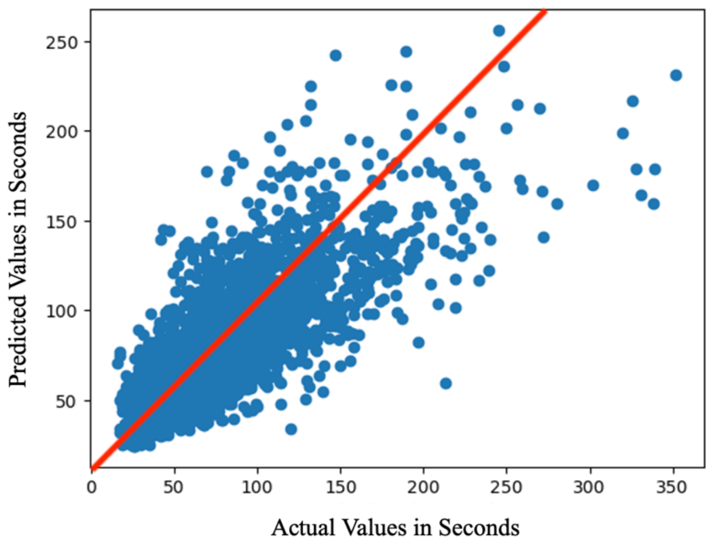


Fig. 4.6: Scatter plot of KNN

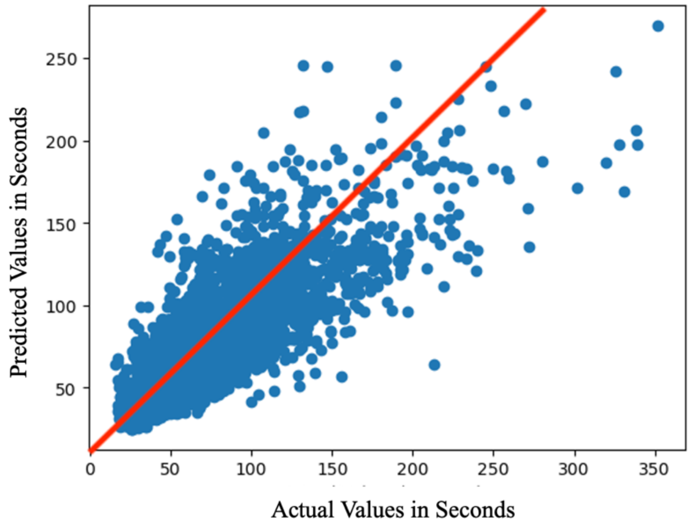


Fig. 4.7: Scatter Plot of AdaBoost

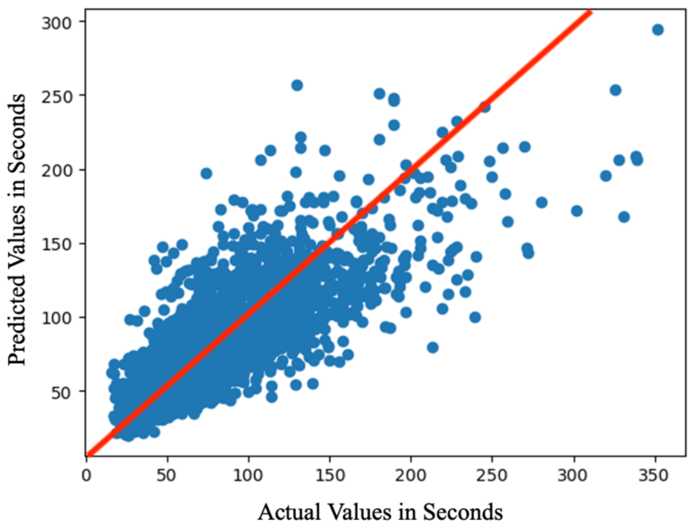


Fig. 4.8: Scatter Plot of XGBoost

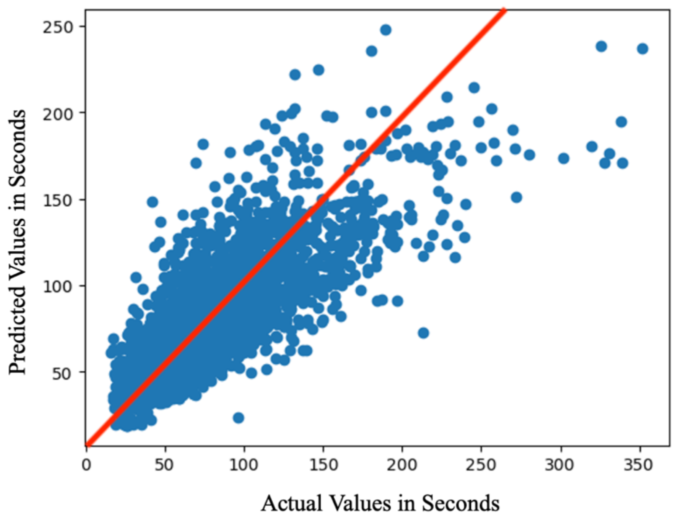


Fig. 4.9: Scatter Plot of GBM

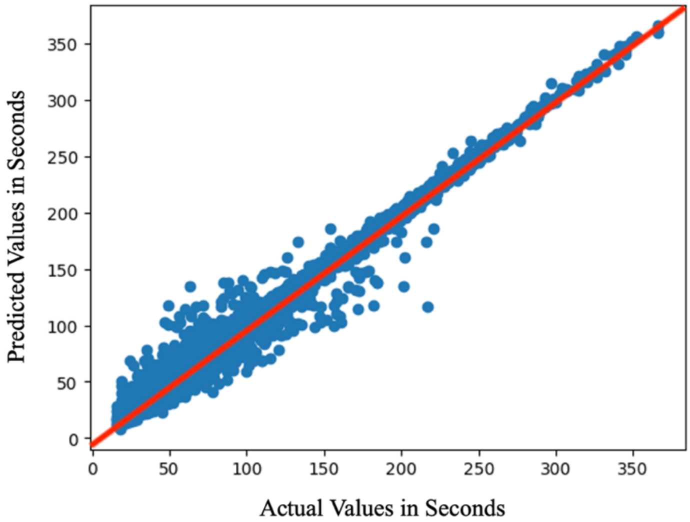


Fig. 4.10: Scatter Plot of LSTM

Evaluation of the best-suited model for travel time prediction was done by comparing the evaluation parameters used. A comparison of the obtained evaluation parameters after running all the models is stated in Table 4.16.

TABLE 4.16: COMPARISON OF MODEL EVALUATION PARAMETERS

Model	**MAE**	**MSE**	**RMSE**	**R Square**
SVR	17.717	669.587	25.876	0.618
RFR	17.283	611.299	24.724	0.651
KNN Regression	18.145	673.327	25.949	0.616
Ada Boost	17.458	627.056	25.041	0.642
XG Boost	17.446	630.786	25.115	0.640
GBM	17.397	618.265	24.865	0.647
LSTM	3.826	36.028	6.002	0.983

When evaluating traditional machine learning models, Random Forest Regression (RFR) outperformed K-Nearest Neighbors (KNN), achieving a Mean Absolute Error (MAE) of 17.3 and an R2R2 value of 0.651. In contrast, the KNN model showed lower prediction accuracy, with an MAE of 18.15. Among the three boosting models, XGBoost, AdaBoost, and Gradient Boosting Machine (GBM), XGBoost and AdaBoost demonstrated comparable performance, with MAE values of 17.45 and 17.46, respectively, and an R2R2 value of 0.64. However, GBM slightly outperformed both, with an MAE of 17.4 and an R2R2 value of 0.647.

The Support Vector Regression (SVR) model showed reasonable accuracy with an R2R2 value of 0.62, but its error metrics were less favourable compared to RFR and GBM.

The Long Short-Term Memory (LSTM) model significantly outperformed all other models, achieving the lowest MAE of 3.86 and the highest R2R2 value of 0.98, indicating that it explains 98.3% of the variance. Due to its superior performance, the evaluation parameters were plotted separately for the LSTM model in Figure 4.11.

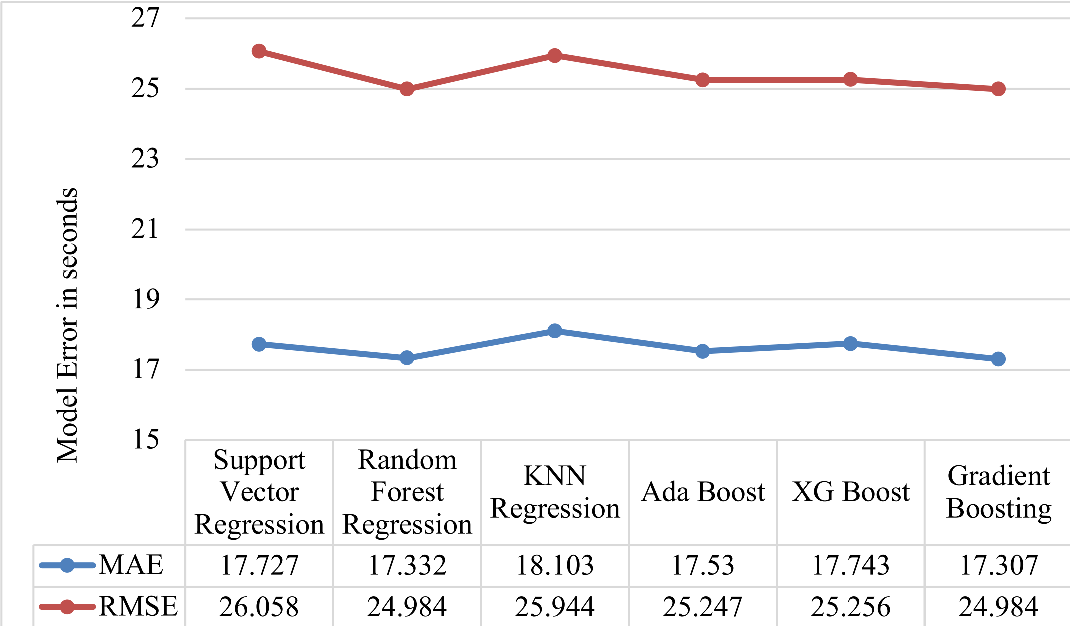


Fig. 4.11: Comparison of MAE and RMSE values of selected models

It was observed that the Gradient Boosting model had the lowest MAE value. Both models Random Forest Regression and Gradient Boosting model performed nearly identically giving the same RMSE value.

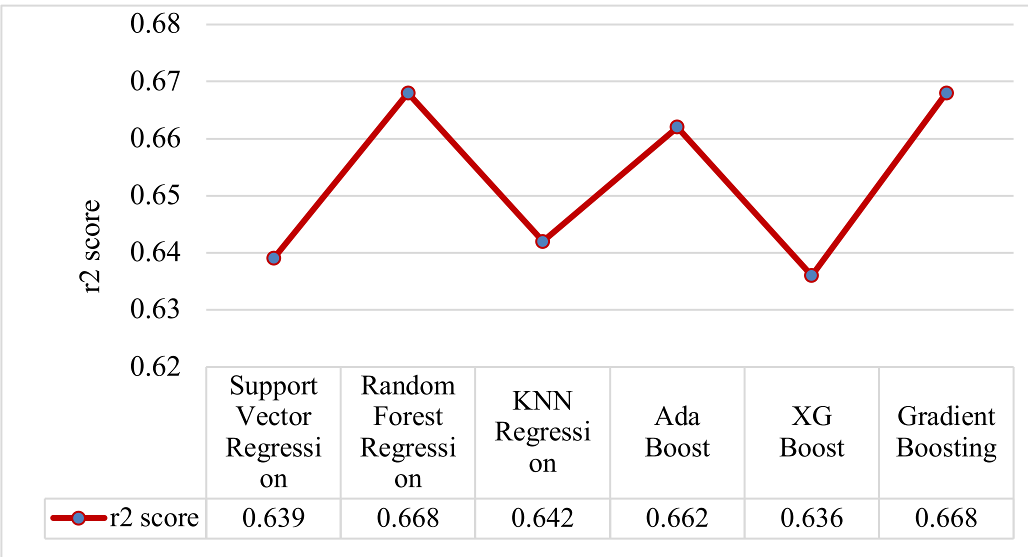


Fig. 4.12: Comparison of R2 score of Selected Models

As shown in Figure 4.12 the highest r^2 score value is given by the Random Forest Regression model and Gradient Boosting models. Although the above models performed considerably well when compared with the models developed in previous research, LSTM which is a deep learning model was also tried out considering the time series data collected using the GPS model. The results of the LSTM model were as follows. The main evaluation parameter that was used in the research was MAE, as it is a direct measure of the forecasting accuracy of the model presented with the same units as the data. Comparison of MAE values of all developed models is shown in Figure 4.13.

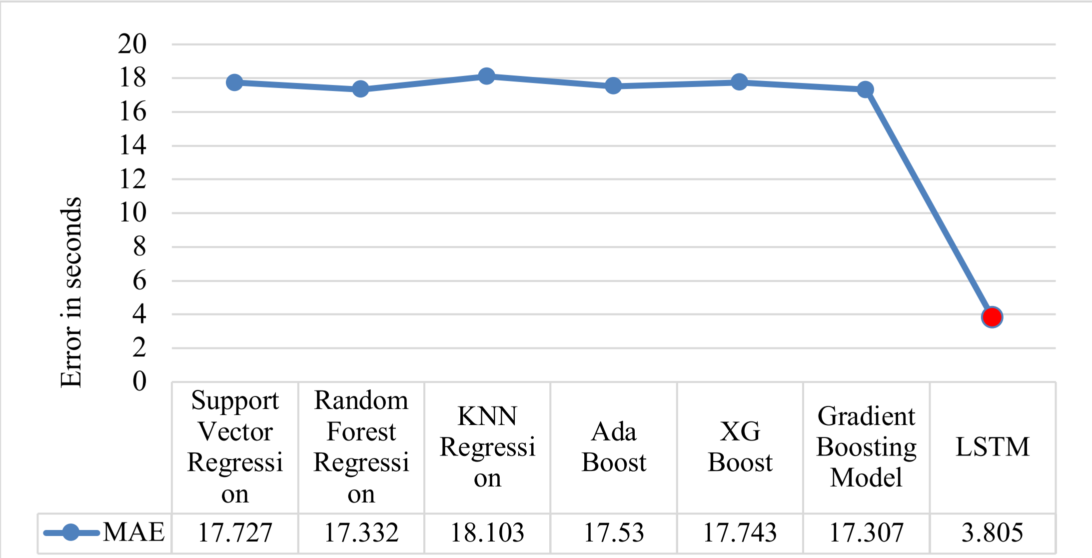


Fig. 4.13: Comparison of MAE values of all Developed Models

The MAE was reduced to a lower value of 3.8 seconds when LSTM was used on the data set, making it the best-performing model out of all by a greater margin.

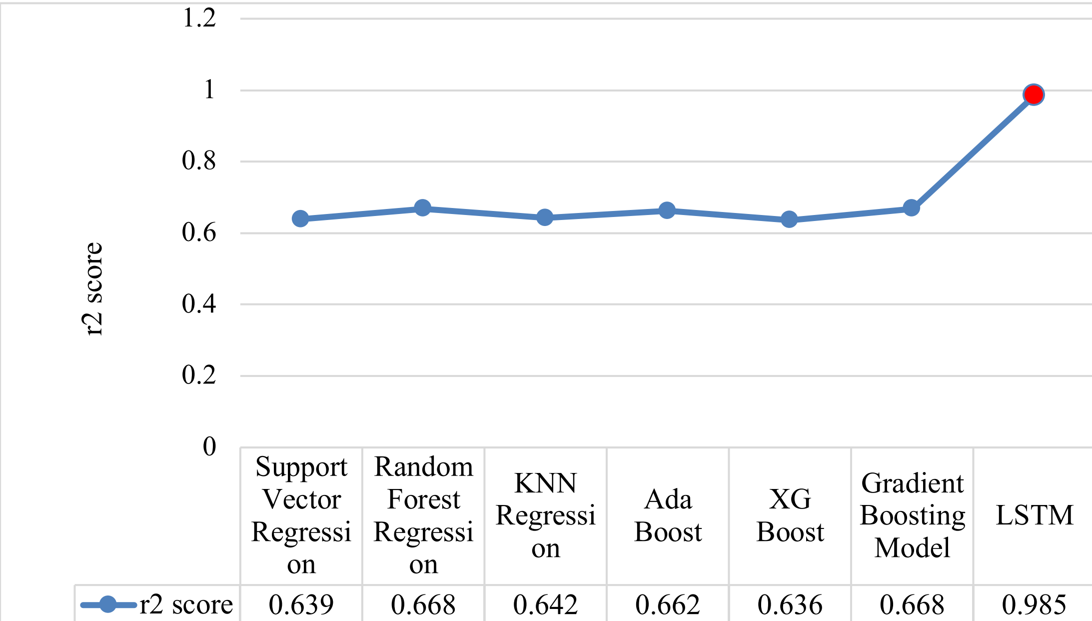


Fig. 4.14: Comparison of R2 Score of All Developed Models

The r2 score of all models is shown in Figure 4.14, and it shows a drastic increase with LSTM when compared to other models, with a value of 0.985, which indicates that 98.5% of the variance in the dependent variable could be explained by the independent variable in the model.

The above models were run on a complete data set having data covering all variables and different categories considered in those variables. However, when applying the above model to other bus routes, it is needed to identify what data variations were less prominent to narrow down the related data collections. Therefore, the final selected model was run with different categories of the date variable selected, narrowing the date to weekdays and weekends, instead of separate seven weekdays. The above data set reduced the MAE of the model to 5.8 seconds, which is a 2-second difference from the MAE of the model with more categorized day data.

4.4 Model Validation

To validate the selected model, a new data file of one bus turn data was used. After reloading the saved pickle files, the above data set is loaded without the travel time, and the travel times are predicted. The comparison of the travel time values predicted by the model, and the actual values of travel time data to bus stops is shown in Table 4.17 below.

TABLE 4.17: ACTUAL TRAVEL TIME VS PREDICTED TRAVEL TIME

Bus Stop Section	Actual Values of Travel Time in seconds	Predicted Values of Travel Time in seconds	Difference in Value	Percentage Difference
1MC	28	34	6	19%
2MC	145	155	10	7%
3MC	41	39	2	5%
4MC	34	34	0	0%
5MC	152	151	1	1%
6MC	70	69	1	1%
7MC	58	66	8	13%
8MC	48	43	5	11%
9MC	46	44	2	4%
10MC	30	38	8	24%
11MC	61	64	3	5%
12MC	115	108	7	6%
13MC	31	31	0	0%
14MC	31	42	11	30%

15MC	271	268	3	1%
16MC	26	32	6	21%
17MC	37	48	11	26%
18MC	33	43	10	26%
19MC	33	37	4	11%
20MC	69	70	1	1%
21MC	92	91	1	1%
22MC	67	63	4	6%
23MC	97	96	1	1%
24MC	86	87	1	1%
25MC	41	43	2	5%
26MC	73	74	1	1%
27MC	33	29	4	13%
28MC	77	76	1	1%
29MC	42	45	3	7%
30MC	88	87	1	1%
31MC	91	89	2	2%
32MC	220	225	5	2%
33MC	49	49	0	0%
34MC	76	78	2	3%
35MC	62	63	1	2%
36MC	99	91	8	8%
37MC	85	81	4	5%
38MC	155	161	6	4%
39MC	52	49	3	6%
40MC	60	69	9	14%
41MC	48	38	10	23%
42MC	57	61	4	7%
43MC	174	171	3	2%
44MC	107	103	4	4%
45MC	70	79	9	12%
46MC	13	9	4	36%
47MC	79	89	10	12%
48MC	46	46	0	0%
49MC	32	24	8	29%
50MC	80	89	9	11%
51MC	111	115	4	4%
52MC	99	95	4	4%
Total trip time	**3,920**	**3,981**	**61**	**2%**

The bus travel times to bus stops were predicted accurately, giving deviations below 11 seconds for each bus route section. The total travel time of the actual trip was recorded as 3,920 seconds, while the predicted total travel time was 3,981, giving a 61-second difference for the total trip, which was a 2% difference of the total travel time.

4.5 Model Applications

The methodology developed in this study is adaptable for application to any short-distance bus route, provided that the requisite data is properly collected and integrated into the model framework. This research identifies the most accurate predictive model configuration, including optimal parameter values and scaling techniques, representing a novel contribution distinct from prior work. The following procedural steps are recommended for implementing the model on a different bus route:

Step 1: Data Collection

Obtain travel time data for the selected bus route over a 30-day period using GPS-enabled devices. To mitigate the data inconsistencies encountered in this study, GPS devices should be installed in a fixed, low-vibration location within each bus. Additionally, training bus operators to mark arrival times at designated stops can significantly reduce the burden of data preprocessing. It is also necessary to collect supplementary data relevant to the model, including road infrastructure details and weather conditions.

Step 2: Model Adjustment

Modify the model's architecture to correspond to the specific number of bus stops on the new route. While the prototype was based on 52 bus stops, the model is scalable and can be adapted according to the number of bus stops where the travel times should be predicted.

Step 3: Model Execution

Execute the model using a consistent train-test data split strategy to ensure robust performance evaluation. Proceed with generating travel time predictions based on the input features collected.

The dataset used in this study was obtained from Route No. 100, which operates between Colombo and Moratuwa, one of the busiest urban bus corridors in Sri Lanka. This route was selected because it captures a wide variety of traffic and operational conditions typical of metropolitan areas, including frequent signalized intersections, heterogeneous traffic flows, and high passenger demand. Therefore, the travel patterns and delays observed along this route are considered representative of urban bus operations within the Western Province.

While the dataset is geographically specific, the methodological framework developed based on GPS-derived travel times and LSTM-based prediction modelling is not limited to this corridor. The same approach can be applied to other routes or cities, provided that real-time positional data and corresponding operational parameters are available. The time frame of the collected data also included both peak and off-peak hours, ensuring that temporal variations in traffic conditions were adequately represented. Thus, the dataset is both temporally and spatially representative for demonstrating the applicability of the proposed model. Future studies can extend this methodology to intercity or rural routes by retraining the model with route-specific data to ensure localized calibration and performance consistency.

To deploy the developed LSTM bus arrival time prediction model in a real-time operational environment, a continual learning approach can be incorporated to allow the model to adapt dynamically to evolving traffic conditions and service patterns. In this approach, the model would be periodically updated with newly collected GPS and operational data without requiring complete retraining from scratch. Techniques such as incremental learning or online learning can be used to fine-tune model weights based on recent observations while preserving previously learned patterns. This would enable the model to respond to changes such as traffic disruptions, seasonal variations, or route modifications, maintaining prediction accuracy over time. Additionally, mechanisms like sliding window training or replay buffers can be implemented to retain a representative subset of historical data, preventing the model from forgetting previously learned traffic patterns a phenomenon known as catastrophic forgetting. By integrating continual learning, the LSTM model can provide up-to-date bus arrival time predictions for passengers and transit operators, ensuring its practical applicability in real-time Intelligent Transport Systems.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions

In this study, as identified in the literature, many research work has been carried out relative to different bus travel time predictions. Historical average models, regression models, Kalman filtering technique and machine learning techniques are identified as models which were incorporated in previous studies. Due to the development of computer-based algorithms, machine learning has come into contact with many areas due to its processing power and increased efficiency and automation. This study mainly focused on the machine learning models used in bus travel time predictions. Through the literature review, ML models SVR, KNN, RF, GBM, AdaBoost, XG Boost and LSTM were identified as the most compatible models.

The data for the travel time prediction model was collected from GPS devices. The study area in this research was the 100 bus route from Moratuwa – Colombo, which contained 53 bus stops. Complete data of 320 bus turns was collected covering 30 days, and the data was processed after filtering travel time data to each bus stop from the GPS data. The factors affecting bus travel time were identified through literature as the starting bus stop, ending bus stop, number of signalized intersections, number of signalized crossings, hour of the day, day of the week, distance travelled, weather conditions, availability of bus lanes and previous travel times. Data was collected under each of the above variables.

Using the above dataset, the selected travel time predictions were run and were evaluated using the model evaluation parameters, MAE, MSE, MAPE, r2 score and RMSE. Out of the tested models, SVR, RFR, KNN, AdaBoost, XGBoost and GBR performed similarly to each other, showing minute changes in their MAE when considering the travel time between two bus stops. It was clearly evident that the LSTM model performed significantly better than the other models due to the time series nature of the data set, providing the lowest MAE. Therefore, the LSTM model using the standard scaler method can be concluded as the best model out of all seven models tested in this study.

When conducting the data collection, several challenges were faced. The data in this research were summarized manually, which was time-consuming. Familiarizing bus operators with the GPS devices and letting them put special GPS points when they arrive at bus stops will reduce the time taken for data filtering, letting the researchers identify the bus arrival times at each bus stop directly. Afterwards, relevant route characteristics, weather data and road characteristics data such as bus lane, signalized intersections and crossings data are required for all selected variables identified. After formulating the data set, it can be loaded into the above model. The data set should be split into lists according to the number of bus stops in the route to fit the model to the changed data set.

Another issue faced during data collection was the inaccurate data points given by the GPS machines. When GPS devices were stored in buses for data collection, severe vibrations of the buses when they were at a stop created inaccurate data points, leading to many bus turn data points not making it to the data set. It recorded data points which were completely away from the actual bus route. Therefore, GPS devices should be stored in places on buses with minimal vibration. The lack of variation in the weather variable within the dataset is another drawback of the study. Most data collection days were predominantly sunny, resulting in limited weather variability. This could have minimized the impact of the weather variable on travel time predictions. Therefore, obtaining a dataset with a broader range of weather conditions would likely improve the model.

The methodology established in this study can be applied to other bus routes with minimal adjustments. To achieve comparable levels of predictive accuracy, it is recommended that travel time data be collected over a minimum period of 30 days for each segment of the route using GPS tracking devices. While the data collection process may be simplified by aggregating individual days into two categories weekdays and weekends. However, such a modification may lead to a slight degradation in model performance, potentially increasing the average prediction error by approximately 05 seconds.

When applying the selected model to alternative bus routes, several critical considerations must be addressed. The model was trained to capture the bus travel and traffic flow patterns present along the original route during the data collection period. Consequently, the model assumes that similar traffic conditions persist during its application. If substantial changes occur in the traffic dynamics of a given route such as due to infrastructure modifications, new traffic regulations, or construction activities model performance may degrade. In such cases, it is advisable to initiate a new round of data collection to reflect the updated traffic conditions, followed by retraining the model to ensure predictive accuracy under the new flow patterns.

An additional limitation of this study was the treatment of traffic flow data. Due to the unavailability of quantitative traffic volume measurements, traffic flow was set as a categorical variable, broadly classified into peak and off-peak hours. However, this categorization exhibited a strong correlation with the 'hour of the day' variable, resulting in its exclusion from the final model. This change enhanced the accuracy by allowing the model to independently learn time-of-day variations in travel time.

Nevertheless, in the event of significant changes where major deviations occurred in a route section that changed its peak hours, a practical solution would be to realign the time-based features, redefining and shifting the hours that are considered peak based on the updated traffic patterns observed during a subsequent data collection phase.

A notable limitation of this study is the inability to link the developed model with real-time data, due to infrastructure constraints. In an ideal scenario where buses are equipped with GPS tracking systems, the model could be integrated with real-time data, enabling dynamic adjustments to predictions based on the bus's current location. This integration would allow the model to provide more accurate predictions by continuously aligning travel time forecasts with real-time conditions.

Another limitation arises from the model's inability to account for sudden, unforeseen changes in traffic flow, such as those caused by road blockages or accidents. Since the model was trained solely on historical datasets, it cannot respond to these real-time disruptions. However, incorporating real-time data would improve the model's performance in such situations, as it would adjust predictions dynamically based on the bus's location and prevailing traffic conditions. In this way, more accurate travel time predictions could be generated, even when the bus is in motion and facing unexpected disruptions.

5.2 Recommendations

The study conducted could be expanded to many more directions for future research. The same methodology and model could be applied to long-distance bus routes by omitting some variables from the selected variables by doing a preliminary data collection. Another major improvement which could be made to the travel time predictions done by the model is incorporating and merging real-time data with the above model. This will reduce the cumulative errors that occurred after the bus had left its origin. It will also identify delays that occurred due to accidents or any other unforeseen circumstances at a route stretch and will notify the passengers at the upcoming bus stops about the delay that occurred.

The above research identifies the machine learning models that could be used for short-term bus travel time predictions while highlighting which models performed better than the others. It identifies a highly accurate deep learning model LSTM for travel time predictions and verifies its predictions for future use.

REFERENCES

[1] K. N. ab and A. H. Abdullah, “A survey on intelligent transportation systems,” *Middle East J Sci Res*, vol. 15, no. 5, pp. 629–642, 2013, doi: 10.5829/idosi.mejsr.2013.15.5.11215.

[2] M. Paradowska, “Intelligent transport systems as an instrument for sustainable development,” *Economic and Environmental Studies*, vol. 11, no. 4, pp. 389–403, 2011.

[3] R. Mangiaracina, A. Perego, G. Salvadori, and A. Tumino, “A comprehensive view of intelligent transport systems for urban smart mobility,” Jan. 02, 2017, *Taylor and Francis Ltd.* doi: 10.1080/13675567.2016.1241220.

[4] T. Kanagasabai, N. Nishanthan, K. Thiruthanigesan, and P. Georgakis, “Towards the development of intelligent transportation systems in sri lanka,” *Article in International Journal of Scientific & Technology Research*, vol. 6, 2017, [Online]. Available: www.ijstr.org

[5] N. Servos, X. Liu, M. Teucke, and M. Freitag, “travel time prediction in a multimodal freight transport relation using machine learning algorithms,” *Logistics*, vol. 4, no. 1, Mar. 2020, doi: 10.3390/logistics4010001.

[6] P. Kumar and S. Suman, “Public transportation systems with its technologies,” in *Proceedings of civil engineering conference- innovation without limits*, Sep. 2009.[Online].Available: https://www.researchgate.net/publication/344780710

[7] R. G. Mishalani, S. Lee, and M. R. McCord, “Evaluating real-time bus arrival information systems,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1731, no. 1, pp. 81–87, Jan. 2000, doi: 10.3141/1731-10.

[8] A. Millonig, M. Sleszynski, and M. Ulm, “Sitting, waiting, wishing: waiting time perception in public transport,” in *2012 15th International IEEE Conference on Intelligent Transportation Systems*, IEEE, Sep. 2012, pp. 1852–1857. doi: 10.1109/ITSC.2012.6338777.

[9] L. Perera, N. Jayasooriya, and G. Rajapaksha, “Assessment of park and ride acceptability of users in developing countries using discrete choice experiment,” *Advances in Transportation Studies: an international Journal*, vol. 55, p. 235, 2021.

[10] Z. Gurmu and W. Fan, "Artificial neural network travel time prediction model for buses using only GPS data," *J Public Trans*, vol. 17, no. 2, pp. 45–65, Jun. 2014, doi: 10.5038/2375-0901.17.2.3.

[11] N. Servos, X. Liu, M. Teucke, and M. Freitag, "Travel time prediction in a multimodal freight transport relation using machine learning algorithms," *Logistics*, vol. 4, no. 1, p. 1, Dec. 2019, doi: 10.3390/logistics4010001.

[12] M. Čelan and M. Lep, "Bus-arrival time prediction using bus network data model and time periods," *Future Generation Computer Systems*, vol. 110, pp. 364–371, Sep. 2020, doi: 10.1016/j.future.2018.04.077.

[13] O. Cats and G. Loutos, "Evaluating the added-value of online bus arrival prediction schemes," *Transp Res Part A Policy Pract*, vol. 86, pp. 35–55, Apr. 2016, doi: 10.1016/j.tra.2016.02.004.

[14] H. Zhang, S. Liang, Y. Han, M. Ma, and R. Leng, "A prediction model for bus arrival time at bus stop considering signal control and surrounding traffic flow," *IEEE Access*, vol. 8, pp. 127672–127681, 2020, doi: 10.1109/ACCESS.2020.3004856.

[15] G. S. Galappaththi and S. Lanka, "Estimating travel time for un-signalized two lane highways in sri lanka," University of Moratuwa, Moratuwa, 2016.

[16] R. Jeong and L. R. Rilett, "Bus arrival time prediction using artificial neural network model," in *IEEE Intelligent Transportation Systems Conference*, Washington D.C., Oct. 2004, pp. 988–993.

[17] L. Vanajakshi, S. C. Subramanian, and R. Sivanandan, "Travel time prediction under heterogeneous traffic conditions using global positioning system data from buses," *IET Intelligent Transport Systems*, vol. 3, no. 1, pp. 1–9, 2009, doi: 10.1049/iet-its:20080013.

[18] D. N. D. Jayaratne, C. J. Vidanapathirana, and H. R. Pasindu, "Challenges faced in heterogeneous traffic data collection: A comparison of traffic data collection technologies," *J Natl Sci Found*, vol. 48, no. 3, pp. 227–237, 2020, doi: 10.4038/jnsfsr.v48i3.8803.

[19] B. Dhivyabharathi, B. A. Kumar, and L. Vanajakshi, "Real-time bus arrival time prediction system under Indian traffic conditions," in *2016 IEEE International Conference on Intelligent Transportation Engineering (ICITE)*, IEEE, Aug. 2016, pp. 18–22. doi: 10.1109/ICITE.2016.7581300.

[20] N. C. Petersen, F. Rodrigues, and F. C. Pereira, “Multi-output bus travel time prediction with convolutional LSTM neural network,” *Expert Syst Appl*, vol. 120, pp. 426–435, Apr. 2019, doi: 10.1016/j.eswa.2018.11.028.

[21] N. Julio, R. Giesen, and P. Lizana, “Real-time prediction of bus travel speeds using traffic shockwaves and machine learning algorithms,” *Research in Transportation Economics*, vol. 59, pp. 250–257, Nov. 2016, doi: 10.1016/j.retrec.2016.07.019.

[22] S. J. Barbeau, “Quality control - lessons learned from the development and evaluation of GTFS-realtime feeds,” in *97th Annual Meeting of the Transportation Research Board*, Washington DC, Nov. 2017.

[23] A. M. Abdelfattah and M. K. Ata, “Models for predicting bus delays,” *Transportation Research Record*, vol. 1623, no. 1, pp. 8–15, 1998.

[24] D. Panovski and T. Zaharia, “Long and short-term bus arrival time prediction with traffic density matrix,” *IEEE Access*, vol. 8, pp. 226267–226284, 2020, doi: 10.1109/ACCESS.2020.3044173.

[25] L. Ye, P. Thiengburanathum, and P. Thiengburanathum, “A conceptual framework for bus arrival prediction based on spark framework and machine learning approaches,” in *2020 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON)*,IEEE,Mar.2020,pp.216–221.doi: 10.1109/ECTIDAMTNCON48261.2020.9090693.

[26] M. Altinkaya and M. Zontul, “Urban Bus Arrival Time Prediction: A review of computational models,” *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 2, no. 4, pp. 164–169, 2013.

[27] O. Eriksson and F. Andres, “Intelligent transport systems and services supported by standards,” in *9th World Congress on Intelligent Transport Systems*, Chicago, Oct. 2002.

[28] M. L. Mfenjou, A. A. Abba Ari, W. Abdou, F. Spies, and Kolyang, “Methodology and trends for an intelligent transport system in developing countries,” *Sustainable Computing: Informatics and Systems*, vol. 19, pp. 96–111, Sep. 2018, doi: 10.1016/j.suscom.2018.08.002.

[29] A. E. Coronado Mondragon, C. S. Lalwani, E. S. Coronado Mondragon, C. E. Coronado Mondragon, and K. S. Pawar, “Intelligent transport systems in multimodal logistics: A case of role and contribution through wireless vehicular

networks in a sea port location," *Int J Prod Econ*, vol. 137, no. 1, pp. 165–175, May 2012, doi: 10.1016/j.ijpe.2011.11.006.

[30] W. H. K. Lam, "Special issue: Intelligent transportation systems," *J Adv Transp*, vol. 36, no. 3, pp. 225–229, Sep. 2002, doi: 10.1002/atr.5670360302.

[31] N. R. Velaga, M. Quddus, and A. L. Bristow, "Detecting and correcting map-matching errors in location-based intelligent transport systems," in *12th World Conference on Transport Research Society (WCTRS)*, Jul. 2010. [Online]. Available: https://www.researchgate.net/publication/229019661

[32] S. Stephenson, T. Moore, T. Edwards, X. Meng, and A. Baxendale, "Accuracy requirements and benchmarking position solutions for intelligent transportation location-based services," in *Proceedings of the 8th International Symposium of Location- Based Services*, Nov. 2011. [Online]. Available: https://www.researchgate.net/publication/260000200

[33] M. Wigan and R. Clarke, "Transport and surveillance aspects of location-based services," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2105, no. 1, pp. 92–99, Jan. 2009, doi: 10.3141/2105-12.

[34] D. H. Stojanovic and S. J. Djordjevic-Kajan, "Developing location-based services from a GIS perspective," in *5th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Service. TELSIKS 2001. Proceedings of Papers (Cat. No.01EX517)*, IEEE, pp. 459–462. doi: 10.1109/TELSKS.2001.955819.

[35] A. Dey, "Location-based services," in *A Dictionary of Transport Analysis*, CRC Press, 2010, pp. 247–249. doi: 10.1201/9781439812273-18.

[36] M. J. W. A. Vanderschuren, "The benefits of intelligent transport systems: modelling the effects of different its systems," Pretoria, 2003.

[37] M. J. W. A. Vanderschuren, "Intelligent transport systems for south africa : impact assessment through microscopic simulation in the south african context," 2006.

[38] K. N. Qureshi and A. H. Abdullah, "A survey on intelligent transportation systems," *Middle East J Sci Res*, vol. 15, no. 5, pp. 629–642, 2013, doi: 10.5829/idosi.mejsr.2013.15.5.11215.

[39] A. Jarašūniene, "Research into intelligent transport systems (ITS) technologies and efficiency," *Transport*, vol. 22, no. 2, pp. 61–67, Jan. 2007, doi: 10.1080/16484142.2007.9638100.

[40] Y. Xing *et al.*, “Identification and analysis of driver postures for in-vehicle driving activities and secondary tasks recognition,” *IEEE Trans Comput Soc Syst*, vol. 5, no. 1, pp. 95–108, 2017.

[41] R. Labayrade, “How autonomous mapping can help a road lane detection system ?,” in *9th International Conference on Control, Automation, Robotics and Vision*, IEEE, 2006, pp. 1–6. doi: 10.1109/ICARCV.2006.345202.

[42] Z. Tao, P. Bonnifait, V. Fremont, and J. Ibanez-Guzman, “Mapping and localization using GPS, lane markings and proprioceptive sensors,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, Nov. 2013, pp. 406–412. doi: 10.1109/IROS.2013.6696383.

[43] M. Shahgholian and D. Gharavian, “Advanced traffic management systems: an overview and a development strategy,” *Electrical Engineering and Systems Science*, Oct. 2018.

[44] A. B. Nkoro and Y. A. Vershinin, “Current and future trends in applications of Intelligent Transport Systems on cars and infrastructure,” in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, IEEE, Oct. 2014, pp. 514–519. doi: 10.1109/ITSC.2014.6957741.

[45] A. Linder, A. Kircher, A. Vadeby, and S. Nygardhs, “Intelligent Transport Systems (ITS) in passenger cars and methods for assessment of traffic safety impact: a literature review,” Linkoping, 2007.

[46] G. Giannopoulos, E. Mitsakis, J. M. Salanova, Dilara. P, P. Bonnel, and V. Punzo, “Overview of Intelligent Transport Systems (ITS),” in *JRC Scientific and Policy Reports* , vol. 1, 2012.

[47] M. Lu *et al.*, “Cooperative and connected intelligent transport systems for sustainable european road transport,” in *Transport Research Arena (TRA)*, Apr. 2018.

[48] S. Kurihara, H. Tamaki, M. Numao, J. Yano, K. Kagawa, and T. Morita, “Traffic congestion forecasting based on pheromone communication model for intelligent transport systems,” in *2009 IEEE Congress on Evolutionary Computation*, IEEE, May 2009, pp. 2879–2884. doi: 10.1109/CEC.2009.4983304.

[49] T. Binjammaz, A. Al-Bayatti, and A. Al-Hargan, “GPS integrity monitoring for an intelligent transport system,” in *2013 10th Workshop on Positioning, Navigation and Communication (WPNC)*, IEEE, Mar. 2013, pp. 1–6. doi: 10.1109/WPNC.2013.6533268.

[50] P. Boccardo, F. Arneodo, and D. Botta, "Application of geomatic techniques in infomobility and intelligent transport systems (ITS)," *Eur J Remote Sens*, vol. 47, no. 1, pp. 95–115, Apr. 2014, doi: 10.5721/EuJRS20144707.

[51] B. Kádár and M. Gede, "Where do tourists go? Visualizing and analysing the spatial distribution of geotagged photography," *Cartographica*, vol. 48, no. 2, pp. 78–88, Jan. 2013, doi: 10.3138/carto.48.2.1839.

[52] W. H. K. Lam, "Development of intelligent transport systems in Hong Kong," in *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No.01TH8585)*, IEEE, pp. 1000–1005. doi: 10.1109/ITSC.2001.948798.

[53] M. Seredynski, D. Khadraoui, and F. Viti, "Signal Phase and Timing (SPaT) for cooperative public transport priority measures," in *22nd ITS World Congress*, Oct. 2015.

[54] N. Hounsell and G. Wall, "New intelligent transport systems applications in europe to improve bus services," *Transportation Research Record : Journal of Transportation Research Board*, vol. 02, pp. 85–91, 1997.

[55] M. Friedrich, F. Leurent, I. Jackiva, V. Fini, and S. Raveau, "From transit systems to models: purpose of modelling," in *Modelling Public Transport Passenger Flows in the Era of Intelligent Transport Systems, Springer Tracts on Transportation and Traffic*, 2016, pp. 131–234. doi: 10.1007/978-3-319-25082-3_4.

[56] O. S. Abejide, J. A. Adedayo, and M. H. Mohamed, "Intelligent transportation system as an effective remedy to improve the public transportation in south africa," 2018.

[57] A. Mewada, S. Tanwar, and Z. Narmawala, "Comparison and evaluation of real time reservation technologies in the intelligent public transport system," in *2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, IEEE, Dec. 2018, pp. 800–805. doi: 10.1109/PDGC.2018.8745888.

[58] S. D. Maclean and D. J. Dailey, "Busview: a graphical transit information system," in *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No.01TH8585)*, IEEE, pp. 1073–1078. doi: 10.1109/ITSC.2001.948811.

[59] B. Ferris, K. Watkins, and A. Borning, "OneBusAway: A transit traveler information system," 2010, pp. 92–106. doi: 10.1007/978-3-642-12607-9_7.

[60] K. Rehrl, N. Göll, S. Leitinger, S. Bruntsch, and H.-J. Mentz, "Smartphone-based information and navigation aids for public transport travelers." [Online]. Available: http://www.vor.at/

[61] I. Jeon, M. Lee, and C. Jun, "A simulation method for adjustment of public transportation routes using geotagged smart card data," 2019, doi: 10.34726/lbs2019.41.

[62] P. Ranasinghe, T. Dammalage, and G. Retscher, "A low-cost public transport tracking and information system for commuters in sri lanka," in *LBS 2019: Adjunct Proceedings of the 15th International Conference on Location-Based Services*, 2019.

[63] L. O. Neverauskiene, M. Novikova, and E. Kazlauskiene, "Factors determining the development of intelligent transport systems," *Business, Management and Economics Engineering*, vol. 19, no. 2, pp. 229–243, 2021.

[64] N. Smith, L. Marquez, and J. Gannon, "Realizing the potential environmental benefits of intelligent transport systems," in *Intelligent Transportation Society of America - 12th World Congress on Intelligent Transport Systems*, Nov. 2009, pp. 2082–2093.

[65] L. Janusova and S. Cicmancova, "Improving safety of transportation by using intelligent transport systems," *Procedia Eng*, vol. 134, pp. 14–12, Jan. 2016.

[66] M. A. Toulouki, E. I. Vlahogianni, and K. Gkritza, "Perceived socio-economic impacts of cooperate intelligent transportation systems; A case study of greek urban road networks," in *5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems*, 2017, pp. 730–737.

[67] B. Kolosz and S. Grant-Muller, "Extending cost–benefit analysis for the sustainability impact of inter-urban intelligent transport systems," *Environ Impact Assess Rev*, vol. 50, pp. 167–177, Jan. 2015, doi: 10.1016/j.eiar.2014.10.006.

[68] A. Stevens, "The applications and limitations of Cost-Benefits Assessments (CBA) for intelligent transport systems," *Reseach in Transportation Economics*, vol. 8, pp. 91–111, 2004.

[69] H. R. Smith, B. Hemily, and M. Ivanovic, *Transit Signal Priority (TSP): A Planning and Implementation Handbook*. 2005.

[70] D. Fayaz, "Intelligent transport system - A review," Electronic Resource.

[71] R. Liu and T. James, “Microsimulation modelling of intelligent speed adaptation systems,” in *PTRC-PUBLICATIONS-P*, 2000, pp. 89–100.

[72] K. Borre, “Global Navigation Satellite Systems (GNSS) and Intelligent Transportation Systems (ITS),” 2019

[73] K. Jaisingh, K. El-Khatib, and R. Akalu, “Paving the way for Intelligent Transport Systems (ITS),” in *Proceedings of the 6th ACM Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications*, New York, NY, USA: ACM, Nov. 2016, pp. 25–31. doi: 10.1145/2989275.2989283.

[74] L. Beer, Z. Mayba, and M. Mokonyama, “The use of dracula microscope modelling in the evaluation of public transport priority measures on the soweto to parktown corridor,” in *24th Annyal South African Transport Conference*, Pretoria, 2005, pp. 11–13.

[75] S. Elkosantini and S. Darmoul, “Intelligent public transportation systems: a review of architectures and enabling technologies,” in *2013 International Conference on Advanced Logistics and Transport*, IEEE, May 2013, pp. 233–238. doi: 10.1109/ICAdLT.2013.6568465.

[76] J. Roux and C. Bester, “The performance of the high occupancy vehicle near Cape Town,” in *21st Annual South African Transport Conference*, Pretoria (SA), 2002.

[77] C. H. Wu, J. M. Ho, and D. T. Lee, “Travel-time prediction with support vector regression,” in *IEEE Transactions on Intelligent Transportation Systems*, Dec. 2004, pp. 276–281. doi: 10.1109/TITS.2004.837813.

[78] S. Arhin and R. Z. Zeola, “Transit bus travel time prediction using AVL data; transit bus travel time prediction using AVL data,” *International Journal of Engineering Research and Technology (IJJERT)*, vol. 5, no. 12, pp. 21–26, Dec. 2016, [Online]. Available: www.ijert.org

[79] H. Liu, H. Xu, Y. Yan, Z. Cai, T. Sun, and W. Li, “Bus arrival time prediction based on LSTM and spatial-temporal feature vector,” *IEEE Access*, vol. 8, pp. 11917–11929, 2020, doi: 10.1109/ACCESS.2020.2965094.

[80] S. I.-J. Chien, Y. Ding, and C. Wei, “Dynamic bus arrival time prediction with artificial neural networks,” *J Transp Eng*, vol. 128, no. 5, pp. 429–438, Sep. 2002, doi: 10.1061/(ASCE)0733-947X(2002)128:5(429).

[81] J. Wang, F. Qiao, and J. Lu, “Research and application of the location information in intelligent transportation,” 2013.

[82] R. Md Noor *et al.*, “Predict arrival time by using machine learning algorithm to promote utilization of urban smart bus,”2020, doi: 10.20944/preprints202002.0197.v1.

[83] J. Pan, X. Dai, X. Xu, and Y. Li, “A self-learning algorithm for predicting bus arrival time based on historical data model,” in *2012 IEEE 2nd International Conference on Cloud Computing and Intelligence Systems*, IEEE, Oct. 2012, pp. 1112–1116. doi: 10.1109/CCIS.2012.6664555.

[84] A. Shalaby and A. Farhan, “Prediction model of bus arrival and departure times using AVL and APC data,” *J Public Trans*, vol. 7, no. 1, 2004.

[85] B. M. Williams and L. A. Hoel, “Modeling and forecasting vehicular traffic flow as a seasonal arima process: theoretical basis and empirical results,” *J Transp Eng*, vol. 129, no. 6, pp. 664–672, Nov. 2003.

[86] P. Ranjitkar, L. S. Tey, E. Chakravorty, and K. L. Hurley, “Bus arrival time modeling based on auckland data,” *Transp Res Rec*, vol. 2673, no. 6, pp. 1–9, Jun. 2019, doi: 10.1177/0361198119840620.

[87] J. Patnaik, S. Chien, and A. Bladikas, “Estimation of bus arrival times using APC data,” *J Public Trans*, vol. 7, no. 1, pp. 1–20, Mar. 2004, doi: 10.5038/2375-0901.7.1.1.

[88] C.-T. Lam, B. Ng, and S. H. Leong, “Prediction of bus arrival time using real-time on-line bus locations,” in *2019 IEEE 19th International Conference on Communication Technology (ICCT)*, IEEE, Oct. 2019, pp. 473–478. doi: 10.1109/ICCT46805.2019.8947251.

[89] N. C. Petersen, F. Rodrigues, and F. C. Pereira, “Multi-output deep learning for bus arrival time predictions,” in *Transportation Research Procedia*, Elsevier B.V., 2019, pp. 138–145. doi: 10.1016/j.trpro.2019.09.025.

[90] D. Panovski and T. Zaharia, “Long and short-term bus arrival time prediction with traffic density matrix,” *IEEE Access*, vol. 8, pp. 226267–226284, 2020, doi: 10.1109/ACCESS.2020.3044173.

[91] P. Balasubramanian and R. Rao, “An adaptive long-term bus arrival time prediction model with cyclic variations,” *J Public Trans*, vol. 18, no. 1, pp. 1–18, Mar. 2015, doi: 10.5038/2375-0901.18.1.6.

[92] M. P. D’Angelo, H. M. Al-Deek, and M. C. Wang, “Travel-time prediction for freeway corridors,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1676, no. 1, pp. 184–191, Jan. 1999, doi: 10.3141/1676-23.

[93] Y. Bin, Y. Zhongzhen, and Y. Baozhen, "Bus arrival time prediction using support vector machines," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 10, no. 4, pp. 151–158, Dec. 2006, doi: 10.1080/15472450600981009.

[94] M. Chowdhury, A. Sadek, K. Ma, N. Kanhere, and P. Bhavsar, "Applications of artificial intelligence paradigms to decision support in real-time traffic management," *Transport Research Record*, vol. 1968, no. 1, pp. 92–98, 2006.

[96] O. A. Montesinos López, A. Montesinos López, and J. Crossa, *Multivariate Statistical Machine Learning Methods for Genomic Prediction*. Springer International Publishing, 2022. doi: 10.1007/978-3-030-89010-0.

[97] L. P. Masiero, M. A. Casanova, and M. T. M. de Carvalho, "Travel time prediction using machine learning," in *Proceedings of the 4th ACM SIGSPATIAL International Workshop on Computational Transportation Science*, New York, NY, USA: ACM, Nov. 2011, pp. 34–38. doi: 10.1145/2068984.2068991.

[98] T. Yin, G. Zhong, J. Zhang, S. He, and B. Ran, "A prediction model of bus arrival time at stops with multi-routes," *Transportation Research Procedia*, vol. 25, pp. 4623–4636, 2017, doi: 10.1016/j.trpro.2017.05.381.

[99] B. Yu, W. H. K. Lam, and M. L. Tam, "Bus arrival time prediction at bus stop with multiple routes," *Transp Res Part C Emerg Technol*, vol. 19, no. 6, pp. 1157–1170, 2011, doi: 10.1016/j.trc.2011.01.003.

[100] J. Gong, M. Liu, and S. Zhang, "Hybrid dynamic prediction model of bus arrival time based on weighted of historical and real-time GPS data," in *2013 25th Chinese Control and Decision Conference (CCDC)*, IEEE, May 2013, pp. 972–976. doi: 10.1109/CCDC.2013.6561064.

[101] M. Ali, M. Zaki, I. Ashour, M. Zorkany, and B. Hesham, "Online bus arrival time prediction using hybrid neural network and kalman filter techniques," *International Journal of Modern Engineering Research (IJMER) www.ijmer.com*, vol. 3, [Online]. Available: www.ijmer.com

[102] R. P. S. Padmanaban, K. Divakar, L. Vanajakshi, and S. C. Subramanian, "Development of a real-time bus arrival prediction system for Indian traffic conditions," *IET Intelligent Transport Systems*, vol. 4, no. 3, pp. 189–200, Sep. 2010, doi: 10.1049/iet-its.2009.0079.

[103] Y. Li, C. Huang, and J. Jiang, “Research of bus arrival prediction model based on GPS and SVM,” in *2018 Chinese Control And Decision Conference (CCDC)*, IEEE, Jun. 2018, pp. 575–579. doi: 10.1109/CCDC.2018.8407197.

[104] Y. Chen, M. Guizani, Y. Zhang, L. Wang, N. Crespi, and G. M. Lee, “When traffic flow prediction and wireless big data analytics meet IEEE,” *IEEE Netw*, vol. 33, no. 3, pp. 161–167, 2019.

[105] L. Zhu, S. Shu, and L. Zou, “XGBoost-based travel time prediction between bus stations and analysis of influencing factors,” *Wirel Commun Mob Comput*, vol. 2022, pp. 1–25, Jul. 2022, doi: 10.1155/2022/3504704.

[106] Jinglin Li, Jie Gao, Yu Yang, and Heran Wei, “Bus arrival time prediction based on mixed model,” *China Communications*, vol. 14, no. 5, pp. 38–47, May 2017, doi: 10.1109/CC.2017.7942193.

[107] L. Jian, “Bus arrival time prediction based on Random Forest,” *Advances in Engineering Research (AER) : 5th International Conference on Frontiers of Manufacturing Science and Measuring Technology (FMSMT)*, vol. 130, pp. 867–872, 2017.

[108] Y. Zhang and A. Haghani, “A gradient boosting method to improve travel time prediction,” *Transp Res Part C Emerg Technol*, vol. 58, pp. 308–324, Sep. 2015, doi: 10.1016/j.trc.2015.02.019.

[109] L. Kisgyorgy and L. Rillet, “Travel Time Prediction by Advanced Neural Network,” *Periodica Polytechnica Civil Engineering*, vol. 46, no. 1, pp. 15–32, 2002.

[110] J. Li, “Bus arrival time prediction based on Random Forest,” in *Proceedings of the 2017 5th International Conference on Frontiers of Manufacturing Science and Measuring Technology (FMSMT 2017)*, Paris, France: Atlantis Press, 2017. doi: 10.2991/fmsmt-17.2017.167.

[111] Y. Sun, Q. Yan, Y. Jiang, and X. F. Zhu, “Reliability prediction model of further bus service based on random forest,” *J Algorithm Comput Technol*, vol. 11, no. 4, pp. 327–335, Dec. 2017, doi: 10.1177/1748301817725306.

[112] W. Fang-Jie, W. Fu-Jian, W. Yu-Chen, and C. Bian, “bus travel time prediction based on light gradient boosting machine algorithm,” *Journal of Transportation Systems Engineering and Information Technology*, vol. 19, no. 2, pp. 116–121, Apr. 2019.

[113] Y. Zhong, Y. Shao, and W. Wu, “Short-term traffic flow prediction model based on XG boost,” *Science Technology and Engineering*, vol. 19, no. 30, pp. 337–342, 2019.

[114] Q. Han, K. Liu, L. Zeng, G. He, L. Ye, and F. Li, “A bus arrival time prediction method based on position calibration and LSTM,” *IEEE Access*, vol. 8, pp. 42372–42383, 2020, doi: 10.1109/ACCESS.2020.2976574.

[115] Y. Duan, Y. L.V., and F.-Y. Wang, “Travel time prediction with LSTM neural network,” in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, Nov. 2016, pp. 1053–1058. doi: 10.1109/ITSC.2016.7795686.

[116] T. Yin, G. Zhong, J. Zhang, S. He, and B. Ran, “A prediction model of bus arrival time at stops with multi-routes,” *Transportation Research Procedia*, vol. 25, pp. 4623–4636, 2017, doi: 10.1016/j.trpro.2017.05.381.

[117] S. Ratneswaran and U. Thayasivam, “An improved bus travel time prediction using multi-model ensemble approach for heterogeneous traffic conditions,” in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 2410–2415. doi: 10.1109/ITSC57777.2023.10421794.

[118] A. Munasinghe and D. Dhananjaya, “Bus travel time estimation using gps data: a case-based study for the kandy city,” in *Proceedings of the National Engineering Research Symposium (NERS)*, 2020, pp. 95–102. [Online]. Available: https://www.researchgate.net/publication/354061750

[119] S. Kumarage *et al.*, “Travel time estimation based on dynamic traffic data and machine learning principles,” *IESE Annual Sessions*, pp. 1135–1142, 2017, [Online]. Available: https://www.researchgate.net/publication/354143472

[120] S. Nadeeshan and A. S. Perera, “Multi-step bidirectional LSTM for low frequent bus travel time prediction,” in *MERCon 2021 - 7th International Multidisciplinary Moratuwa Engineering Research Conference, Proceedings*, Institute of Electrical and Electronics Engineers Inc., Jul. 2021, pp. 462–467. doi: 10.1109/MERCon52712.2021.9525709.

[121] S. Oh, Y.-J. Byon, K. Jang, and H. Yeo, “Short-term travel-time prediction on highway: a review of the data-driven approach,” *Transp Rev*, vol. 35, no. 1, pp. 4–32, Jan. 2015, doi: 10.1080/01441647.2014.992496.

[122] R. P. S. Padmanaban, K. Divakar, L. Vanajakshi, and S. C. Subramanian, “Development of a real-time bus arrival prediction system for Indian traffic

conditions," *IET Intelligent Transport Systems*, vol. 4, no. 3, p. 189, 2010, doi: 10.1049/iet-its.2009.0079.

[123] Z. Huang, Q. Li, F. Li, and J. Xia, "A novel bus-dispatching model based on passenger flow and arrival time prediction," *IEEE Access*, vol. 7, pp. 106453–106465, 2019, doi: 10.1109/ACCESS.2019.2932801.

[124] AccuWeather, "Weather forecast for Moratuwa, Sri Lanka." Accessed: Feb. 28, 2022. [Online]. Available: https://www.accuweather.com/en/lk/moratuwa/311405/weather-forecast/311405?city=moratuwa

[125] D. Abbott, *Applied Predictive Analytics - Principles and Techniques for Professional Data Analytics*. John Wiley and Sons Inc, 2014.

[126] F. Lazzeri, *Machine Learning for Time Series Forecasting with Python*. John Wiley & Sons Inc., 2021.

[127] J. Brownlee, *Introduction to Time Series Forecasting with Python*, 1.9. 2020.

Annex A

LSTM Coding

```
import numpy as np
import pandas as pd
import seaborn as sns
import tensorflow as tf
import matplotlib.pyplot as plt
from sklearn.metrics import (
                            mean_squared_error,
                            mean_absolute_error,
                            r2_score,
                            mean_absolute_percentage_error,

                            )
from sklearn.preprocessing import StandardScaler

df = pd.read_csv('LSTM 5.csv')
df.dtypes
weather_dict = {
                'Sunny': 0,
                'Light Rain': 1,
                'Heavy Rain': 2,
                'Moderate Rain': 3
                }
days_dict = {
            'Monday': 0,
            'Tuesday': 1,
            'Wednesday': 2,
            'Thursday': 3,
            'Friday': 4,
            'Saturday': 5,
            'Sunday': 6
            }
bl_dict = {
        'BL': 0,
        'NBL': 1
        }


def split_dataframe():
```

```
    columns = df.columns
    columns = [col.strip() for col in columns]
    df.columns = columns

    indexes = df[df['Start'] == '1MC'].index

    df_list = []
    for i in range(len(indexes)-1):
        df_list.append(df.loc[indexes[i]:indexes[i+1]-1])

    len_df_list = [len(df) for df in df_list]
    len_df_list = pd.Series(len_df_list)
    indexes = len_df_list[(len_df_list == 50) | (len_df_list ==
51) | (len_df_list == 52)].index
    #indexes = len_df_list[(len_df_list == 50)].index
    df_list = [df_list[i] for i in indexes]
    return df_list

def process_single_dataframe(df_sample):
    df_sample = df_sample.drop(['Time', 'Start', 'End'], axis=1)
    df_sample['Weather'] = df_sample['Weather'].map(weather_dict)
    df_sample['Day'] = df_sample['Day'].map(days_dict)
    df_sample['BL'] = df_sample['BL'].map(bl_dict)

    X = df_sample.drop('TT', axis=1).values
    y = df_sample['TT'].values

    X_pad = np.zeros((52, 7))
    y_pad = np.zeros(52)

    X_pad[:X.shape[0], :X.shape[1]] = X
    y_pad[:y.shape[0]] = y

    return X_pad, y_pad

def create_dataset():
    df_list = split_dataframe()
    X = []
    y = []
    for df in df_list:
        X_sample, y_sample = process_single_dataframe(df)
        X.append(X_sample)
        y.append(y_sample)

    X, Y = np.array(X), np.array(y)
```

```
    encoder_x = StandardScaler()
    X = encoder_x.fit_transform(X.reshape(-1, 7)).reshape(-1, 52,
7)

    encoder_y = StandardScaler()
    Y = encoder_y.fit_transform(Y.reshape(-1, 1)).reshape(-1, 52)


    return X, Y, encoder_x, encoder_y


def split_dataframe(df):
    df.columns = [col.strip() for col in df.columns]

    # Identify start indices where 'Start' equals '1MC'
    start_indices = df[df['Start'] == '1MC'].index

    # Initialize empty lists to store segments
    segments = []

    for i in range(len(start_indices) - 1):
        start_idx = start_indices[i]
        end_idx = start_indices[i + 1] - 1
        segment = df.loc[start_idx:end_idx]

        # Only add segments of exactly 50 rows
        if len(segment) == 50:
            segments.append(segment)

    return segments


def split_dataframe(df):
    df.columns = [col.strip() for col in df.columns]

    # Identify start indices where 'Start' equals '1MC'
    start_indices = df[df['Start'] == '1MC'].index

    # Initialize empty lists to store segments
    segments = []

    for i in range(len(start_indices) - 1):
        start_idx = start_indices[i]
        end_idx = start_indices[i + 1] - 1
        segment = df.loc[start_idx:end_idx]
```

```
        # Only add segments of exactly 50 rows
        if len(segment) == 50:
            segments.append(segment)

    return segments

def process_single_dataframe(df_sample):
    df_sample = df_sample.drop(['Time', 'Start', 'End'], axis=1)
    df_sample['Weather'] = df_sample['Weather'].map(weather_dict)
    df_sample['Day'] = df_sample['Day'].map(days_dict)
    df_sample['BL'] = df_sample['BL'].map(bl_dict)

    X = df_sample.drop('TT', axis=1).values
    y = df_sample['TT'].values

    X_pad = np.zeros((50, 7))
    y_pad = np.zeros(50)

    X_pad[:X.shape[0], :X.shape[1]] = X
    y_pad[:y.shape[0]] = y

    return X_pad, y_pad

def create_dataset(df):
    segments = split_dataframe(df)
    X = np.zeros((len(segments), 50, 7))
    y = np.zeros((len(segments), 50))

    for i, segment in enumerate(segments):
        X_sample, y_sample = process_single_dataframe(segment)
        X[i] = X_sample
        y[i] = y_sample

    encoder_x = StandardScaler()
    X = encoder_x.fit_transform(X.reshape(-1, 7)).reshape(-1, 50,
7)

    encoder_y = StandardScaler()
    y = encoder_y.fit_transform(y.reshape(-1, 1)).reshape(-1, 50)

    return X, y, encoder_x, encoder_y


X, Y, encoder_x, encoder_y = create_dataset()
```

```
inputs = tf.keras.Input(shape=(52, 7))
x = tf.keras.layers.LSTM(128, return_sequences=True)(inputs)
x = tf.keras.layers.LSTM(128)(x)

x = tf.keras.layers.Dense(128, activation='relu')(x)
x = tf.keras.layers.Dense(64, activation='relu')(x)
outputs = tf.keras.layers.Dense(52)(x)

model = tf.keras.Model(
                        inputs=inputs,
                        outputs=outputs
                        )

model.compile(
            optimizer='adam',
            metrics=['mae'],
            loss='mse'
            )


history = model.fit(
                  X, Y,
                  epochs=2852,
                  batch_size=16,
                  verbose=1
                  )

# Get the loss values from the history dictionary.
mae = history.history['mae']
mse = history.history['loss']

# Get the epoch numbers.
epochs = range(1, len(mae) + 1)

 # Plot the MAE and MSE values against the epoch numbers.
plt.plot(epochs, mae, label='MAE')
plt.plot(epochs, mse, label='MSE')

# Set the title and axis labels.
plt.title('MAE and MSE for Model Training')
plt.xlabel('Epoch')
plt.ylabel('Loss Value')


P = model.predict(X)
```

```python
P_unscaled = encoder_y.inverse_transform(P)
Y_unscaled = encoder_y.inverse_transform(Y)

P_unscaled = P_unscaled.reshape(-1,)
Y_unscaled = Y_unscaled.reshape(-1,)


mae = mean_absolute_error(Y_unscaled, P_unscaled)
mse = mean_squared_error(Y_unscaled, P_unscaled)
r2 = r2_score(Y_unscaled, P_unscaled)
mape = mean_absolute_percentage_error(Y_unscaled, P_unscaled)

def root_mean_squared_error(Y_unscaled, P_unscaled):
    return np.sqrt(mean_squared_error(Y_unscaled, P_unscaled))

# Example usage:
rmse = root_mean_squared_error(Y_unscaled, P_unscaled)


print("Mean Absolute Error: ", mae)
print("Mean Squared Error: ", mse)
print("R2 Score: ", r2)
print("Mean Absolute Percentage Error: ", mape)
print("Root mean Squared Error: ", rmse)


plt.scatter(Y_unscaled, P_unscaled)

plt.xlabe('Actual Values in seconds')
plt.ylabe("Predicted Values in seconds")
plt.show()
```