Urban Traffic State Estimation Using Probe Vehicles

Thesis submitted in partial fulfilment of the requirements for the award of degree of

Master of Engineering
in
Computer Science and Engineering

Submitted By
Vivek Mehta
(Roll No. 801432030)

Under the supervision of:
Dr. Inderveer Chana
Professor, CSED
Thapar University, Patiala

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

June 2016

Certificate

I hereby certify that the work which is being presented in the thesis titled, "*Urban Traffic State Estimation using Probe Vehicles*", in partial fulfilment of the requirements for the award of degree of Master of Engineering in *Computer Science and Engineering* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of Dr. Inderveer Chana and refers other researchers work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for the award of any other degree of this or any other University.

(Vivek Mehta)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

08/07/16

Dr. Inderveer Chana
Professor,
Computer Science and Engineering Department,
Thapar University, Patiala

Countersigned by

Dr. Maninder Singh
Head,
Computer Science and Engineering Department,
Thapar University, Patiala

Dr. S. S. Bhatia
Dean (Academic Affairs),
Thapar University,
Patiala

Acknowledgement

I pay my lots of love and gratitude to the world Almighty Authority, our **God Father** who at every step remained together to build in me the confidence to complete this work. I pay my deep sense of gratitude to **Dr. Inderveer Chana,** Professor, Computer Science and Engineering Department, Thapar University under strict and sincere guidance of whom, I remained motivated to carry out my work. It was a novel experience to work under her supervision.

I also thank **Dr. Maninder Singh**, Head of Computer Science and Engineering Department (CSED) and **Dr. Ashutosh Mishra**, P.G. Coordinator for their help and encouragement. I would also like to extend my gestures of gratitude to all the staff members and faculty of CSED for their extreme help to build in me the skills required to complete this work especially **Dr. Prashant Singh Rana** who remained always available to help me.

I'm also thankful to all of my classmates knowingly or unknowingly who remained an inspiration for me.

And very importantly, I pay my gratitude to my parents and all the family members for their tremendous support of every kind without which I couldn't see myself at this point.

Vivek Mehta

Vivek Mehta

(801432030)

ii

Abstract

An Intelligent Transportation System (ITS) is a system which uses traffic information to generate useful information about the congestion of road network with the aim of proper utilization of the road network. Traditional techniques to get traffic information relies on fixed road side sensors, inductive loop detectors and video cameras etc. This technique however is not scalable because of high cost of such equipment to be installed on a large road network. An alternate to this technique is the use of Global Positioning System (GPS)-based probe vehicle data. This technique is scalable to a large road network due to the presence of such probe vehicles (taxis). This data can be collected by the use of smart phones.

However, this technique faces the challenge that the data collected through this medium is sparse and unevenly distributed due to random movement of the vehicles throughout the road network. So this kind of data can lead to inaccurate traffic state prediction results. To overcome this challenge, various techniques have been proposed to measure traffic state through this kind of data for which this study provides a literature review along with the research gaps.

In this thesis a model has been proposed to measure the traffic state in the form of average speed of vehicles on a road. The main focus of this research work is to gather the traffic data and present the state of data in an efficient way using Recursive Partitioning and Regression Trees (RPART) machine learning model so that this information can be used for other future developments like urban infrastructure planning, re-routing of traffic etc.

The data used for training and testing the proposed model comes from the simulation of an Indian city road network by the use of traffic simulation software Simulation of Urban MObility (SUMO) and is pre-processed before using it to train the proposed model.

Table of Contents

List of Figures and Tables

Figures

Tables

Chapter 1
Introduction

This chapter gives an introduction to Cloud computing and discusses its important application areas. Along with, it discusses the general details of Intelligent Transportation Systems (ITS) which are used to measure the traffic state of a road network in an urban area and runs as an application on Cloud.

1.1 Cloud computing

Performing the tasks of computation by using the software for development, platforms for deployment, infrastructures for storage, networking, etc. as a service over the internet in a pay per use manner rather than having our own set up is known as Cloud computing. Figure 1.1 shows the Cloud computing concept in which various local users are connected to resources on the Cloud [39]. This kind of elastic infrastructure is needed because of rapid increase of internet connectivity and the requirement of dynamic data accessibility i.e. one can access the data anywhere, anytime on any system.

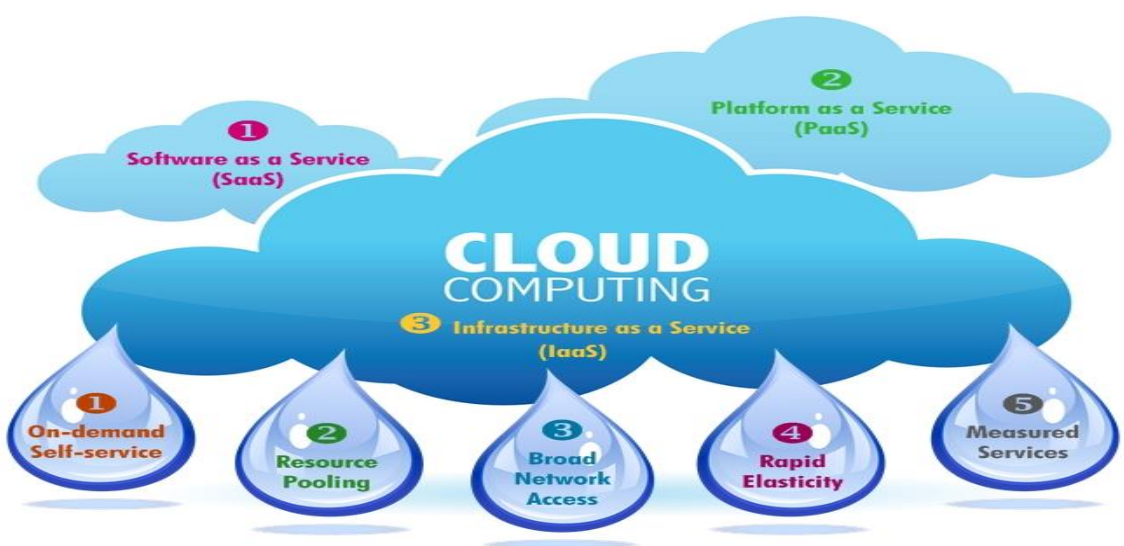

Figure 1.1 Cloud Computing [38]

1.2 Types of Cloud services

The following three types of services are provided by Cloud computing.

1.2.1 Software as a service (SaaS)

This service avails us the use of software and applications that runs on the Cloud operated and maintained by others, and accessed by us through the use of internet. The main advantages of SaaS are:

i. Availability of innovative business applications specific to our business or organization.

ii. Provides the ability to access the data and application from anywhere anytime.

iii. No data lost in case users' system failure as data resides in the Cloud.

iv. Can be dynamically scaled up in accordance to the users' needs.

1.2.2 Platform as a service (PaaS)

This service backs up the complete software development life cycle by provisioning the underlying hardware and software requirements. For example deployment of a new business tool may require new hardware, operating system, middleware etc. These further needs to be maintained by the IT teams with time. So, the PaaS provider takes this overhead to itself and frees the developers from these tasks. The developer just need to log in and start using these software and hardware as a service over the internet through the web browser. Other advantages are:

i. Changing or upgrading the operating system features if needed with time.
ii. It also allows the development teams to collaborate with each other in real time which results in an efficient development process.

1.2.3 Infrastructure as a service (IaaS)

IaaS facilitates the users with resources such as networking hardware, storage, data center space, servers etc. on a pay per use basis. Thus it has the following advantages:

i. Frees the customer from investing in hardware equipment.
ii. Dynamic scalability of the required resource according to the present demand.
iii. Customer gets the benefit of Innovative services that gest available on Cloud with time.

1.3 Cloud Based Application Areas

Some of the important application areas of Cloud computing are listed below and summarized in Figure 1.2.

i. Software Development

Cloud computing allows the software development process to be carried out in a streamlined way. Development assets are made quickly available through Cloud computing. And it also allows the developers, architects and designers to collaborate on development efforts. According to Evans Data Cloud development survey conducted in Dec 2013, Cloud computing reduces on an average the overall development time by 11.6 % [3].

ii. Hospitals and healthcare

To provide quick treatment to the patients, doctors require quick access to the past health records of the patients. Availability of the health records irrespective of the location of both the patient and the clinician can also play a major role in timely treatment and thus producing better treatment results. Traditional way of storing the records cannot provide this facility. Cloud caters to these crucial requirements and allows the access to the patient data anytime anywhere. Thus healthcare organizations are able to access the past health records quickly based upon which they are able to treat or operate the patient well in time.

Development of new drugs require analytics of large datasets like that of DNA sequencing. Pharma organizations can process their large datasets over Cloud in a pay per use manner thus saving them from spending thousands of dollars to setup their own infrastructure for drug research.

Healthcare organizations still continue to depend on client-server systems which are vulnerable to data breaching owing to technology deficiencies, threat etc. as these local servers are placed in poorly secured server rooms. A web-based private Cloud can efficiently manage all these security issues.

iii. Email Communication

Communication through email plays a central role for the people of almost every domain. And this is the area where Cloud computing finds its most frequent and appropriate application. Access of emails through any place is done through Cloud storage itself. To make the email service more reliable all big names like yahoo, Microsoft and google exploits the use of Cloud storage [1].

iv. Local storage up gradation not required

Data stored on our local machines first is limited to amount of storage available and secondly can be accessed by using that particular device only. Online storage on Cloud enables us to access our Mp3s, audios, videos, pictures, documents etc. from anywhere. It also prevents us from the consequences of driver failure. To maintain the integrity of the data Cloud computing implements several layers of security making it nearly impossible to lose our data. A wide range of free and premium Cloud computing solutions are available to choose from. For e.g. Microsoft SkyDrive, Amazon S3, Humyo, Zumodrive etc [2].

v. Collaborative work

Several times the opinion of our peers is needed in real time. Emailing the documents to them and receiving their responses is not an efficient solution. Google wave allows to create a document, invite others to comment or edit them either in real time or whenever they find it suitable thus allowing others to join in the creation of a final draft. Other examples of this kind of Cloud service are Spicebird, mikogo, stixy and vyew [2].

vi. A Virtual Office

Cloud computing services like Google docs, Ajax13, Thinkfree and MS office live are some examples of online suits for office applications like creation of documents, presentations, spreadsheets, pdf files etc. The obvious benefits of using these services are avoiding the storage of heavy programs, improved accessibility and secure storage [21].

vii. Extra processing power

Hiring extra processing power online instead of purchasing extra servers, network equipment etc. can save thousands of dollars to the users. In future if even more processing power is needed can be scaled up as per the requirements. Amazon’s EC2, AbiCloud, Elastichosts, Nebula etc. are the example service providers of this kind [2].

viii. Big data analytics

By exploring the historical data, retailers and suppliers are able to recognize useful buying patterns that help them to advertise and market their product according to a specific segment of population [26]. This historical data appears in the form of both structured and unstructured data and is generally very big in size. Cloud can be leveraged to analyze this huge data to extract useful information which further helps in correct decision making for marketing and advertisement of the products [3].

ix. Disaster recovery

Cloud storage enables disaster recovery in a very smart way from a number of physical locations at a low cost as compared to traditional disaster recovery methods which uses fixed assets and rigid methods at a very high cost [3].

x. Backup

Backing up the data has always remained a tedious and time consuming process involving the manual collection of hard drives and tapes [3]. There is always a possibility of incurring of the problems running out of the media, complex restore operations which are prone to human errors and bugs [27].

Figure 1.2 Cloud based Applications

1.4 Intelligent Transportation System (ITS)

An Intelligent Transportation System (ITS) is basically a system that integrates traffic information, communication and data processing techniques (in real time) for the efficient utilization of roads and thus establish a transportation management system that reduces the overall traffic congestion [28]. Figure 1.3 illustrates the flow of information in an ITS that is based on GPS traffic data for traffic state estimation.

1.4.1 Data acquisition techniques in an ITS

Broadly data acquisition techniques can be classified into following two categories [35]:

i. Fixed road-side sensors

Traditional approaches for data gathering for ITS is done by means of fixed road-side sensors for e.g. loop detectors, video-cameras etc. for the detection of traffic state variables like average speed, travel time, density etc.[18] [19] [20]. However, due to high costs involved or the deployment as well as maintenance, the entire road network cannot be covered by these devices. Also, they cannot reflect a wider view of the speed of a particular vehicle because they measure the speed only at their point of location.

ii. GPS receivers

Alternative approach to fixed sensors is to use GPS receivers which now comes as embedded in vehicles or in smart phones [36]. This technique enables to get parameters like speed, location etc. dynamically on the path of movement of the vehicle (although this happens at random locations). These data updates of speed and location of a vehicle are known as probes and the vehicles in which these GPS devices are embedded are known as probe vehicles (buses, taxis, ambulance etc.).

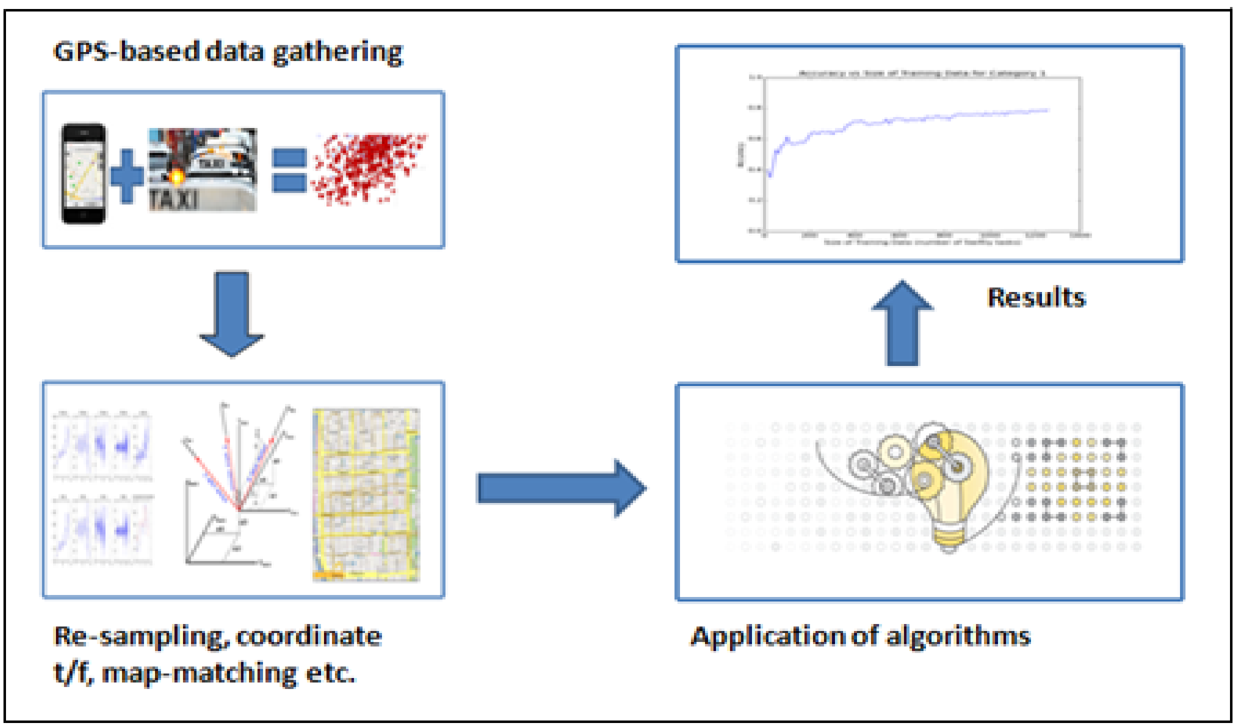

Figure 1.3 Data flow in an ITS

These data probes can be transmitted over a cellular network such as GSM/GPRS to a monitoring center for traffic estimation. As these vehicles cover the entire city the data is collected over a large area and due to the low cost of onboard GPS receivers, the overall cost of the system is low. Figure 1.3 illustrates the flow of data in an ITS.

1.4.2 Traffic State Parameters

A Traffic state parameter reflects the present situation of traffic congestion. Several parameters have been defined in traffic engineering to measure the traffic state for example, travel time, speed, density, flow, occupancy and delay etc. Some important parameters are described as follows [29].

i. Flow rate: It is expressed as number of vehicles per hour, hence is a measurement of traffic state over time.

ii. Mean speed: It is defined as the distance travelled in kilometers per hour. Travel time is correspondingly calculated as length of the road divided by the mean speed.

iii. Density: This defines a spatial measurement of traffic state and is calculated as number of vehicles per kilometer of the road.

In traffic flow theory, these three macroscopic parameters are related as by Eqn 1.1.

$$q = kv \tag{1.1}$$

Where 'q' is the flow rate, 'k' is the density and 'v' is the mean speed.

1.5 Research Motivation

The problem of traffic management is increasing in urban cities of India. Fixed road side sensors such as inductive loop detectors, cameras etc. are not scalable to a large extent due to their high cost. GPS based sensors available in smart phones and on board GPS devices installed in taxis, buses etc. offers a cheaper way to gather traffic information like speed, position and timestamp etc. for a vehicle and deliver it to a server through internet. Thus motivation behind this research is that mining the GPS probe data offers a scalable and less costly way of estimation of traffic state. This traffic state information is of huge importance in managing the traffic and also in urban infrastructure planning. But at the same time, sparse and uneven distribution of probe vehicles throughout the traffic poses a significant challenge towards the probe data processing, thus affecting the accuracy of the results. This study aims at learning the traffic state parameters by proposing an efficient machine learning model that learns from available traffic data and estimates the traffic state parameter (average speed or travel time). Traffic state estimation constitutes an essential part of the future smart cities projects that have been proposed for a large number of cities in India.

1.6 Organization of Thesis

The rest of the thesis is organized in the following way.

- **Chapter 2** describes the survey that has been done for this research. In this chapter the characteristics of an ITS, existing key approaches for traffic state estimation, challenges that need to be handled and finally the existing Cloud based ITS systems have been described.
- **Chapter 3** describes problem statement of traffic state estimation along with the objectives that have been taken for accomplishment for this research.
- **Chapter 4** describes the methodology that have been proposed to achieve the primary aim of this study.
- **Chapter 5** describes the series of experiments that have been involved in achieving the objectives along with the final results.
- **Chapter 6** concludes this research and proposes some work that can be carried out in the future for further improvement in Intelligent Transportation Systems.

Chapter 2

Literature Review

Researchers from both academia and industry have put their efforts for an efficient processing of data involved in an Intelligent Transportation System. Also, several ITS projects have been undertaken to prove the feasibility of such systems for traffic management. This chapter presents various techniques proposed for processing of probe vehicle data along with the existing ITS projects.

2.1 Probe Technology

As already given a glimpse in the introduction, probe technology is an alternative approach to the fixed sensor approach (use of loop detectors, video-cameras etc. to get information related to vehicles) in which either GPS devices equipped in probe vehicles probe vehicles or smart phones are used for getting the traffic information. These data probes can be transmitted over a cellular network such as GSM/GPRS to a monitoring center for traffic estimation [30] [31]. This technology has the advantage of low cost of infrastructure as compared to the cost of implementation of sensors on the complete road network. However, this new approach faces some challenges. Here a survey of the techniques which have been used to overcome these challenges has been presented. One key challenge is that the distribution of the vehicles is uneven and incomplete in-between the whole space and time.

As shown in chapter 1 (Figure 1.3), firstly GPS based vehicles or mobile phones are used to collect data. After which, pre-processing techniques like re-sampling, coordinate transformation, map-matching etc. are applied and then suitable prediction/estimation algorithms are applied to measure the traffic state. To represent the collected GPS measurements precisely on a digital map, coordinate transformation and map matching are used. Re-sampling may also be done on case of synthetic data. Examples of traffic state prediction techniques are Statistical [6], Bayesian, machine learning models etc. [34] [4].

In this study, a comprehensive study of the key techniques proposed in the past recent years for the estimation of traffic state has been presented. This is done by comparing them on the basis of the attributes like accuracy, running time, sampling frequency of the dataset, number of vehicles etc.

2.2 Survey Approach

This survey has been done with an objective to describe the current state of the art in intelligent transportation research using GPS-based probe vehicles for traffic data collection. This has been done by covering the following aspects.

i. Introduction to GPS-based traffic monitoring system and its advantages over traditional fixed sensor based technology.

ii. Characteristics of a traffic monitoring system.

iii. Different approaches that have been used for traffic estimation in the past few years.

iv. Challenges that need to be handled in this technology.

v. Existing projects that have implemented this technology.

2.3 Characteristics of a Traffic Management System

The major factors that characterize a traffic management system are. (originally described by Eleni et al. in [9] and further used by Soufiene et al. in [17].)

i. Scope Determination: Scope determines that whether our forecasting model should be implemented as a part of a traffic management system (TMS) or a traveler information system (ITS). It also determines the area of implementation (e.g. freeway, highway, urban arterials etc.).

ii. Conceptual specification of the output: Here, the size of the horizon and the step [9] has been specified. The forecasting horizon denotes the extent of the time ahead which the traffic state has been predicted for. The forecasting step is the actual time interval upon which traffic state is forecasted and thus gives the frequency of prediction in the forecasting horizon. So, intuitively the larger is the forecasting horizon, lesser will be the accuracy of the model. The shorter is the forecasting step used, the more difficult will be to predict. A 15-min interval of time has been indicated as the best interval by the Highway Capacity Manual (2000).

iii. Methodology for modelling of data: Appropriate technique for traffic forecasting should be made. In 1990s, Autoregressive Integrated Moving Averages (ARIMA) models were used for used for forecasting the urban traffic volume, bottleneck formulation in a freeway etc. but ARIMA models tend to concentrate on means leaving the extremes. Techniques such as Artificial neural network, non-parametric regression are being widely since the last decade.

2.4 Existing Key Approaches for Traffic State Estimation

Some important approaches that have been proposed in the last few years for the measuring the traffic state variables are Bayesian network, Compressive sensing, curve-fitting, statistical, neural-network, K-NN etc. They are summarized as follows.

2.4.1 Bayesian Network Approach

In this approach, firstly a large historical dataset is used to learn about the traffic state variables by using an Expectation-Maximization algorithm. Then traffic state prediction is done in real time with streaming data. Historical training makes the model more robust. Also it has been stated [4] to use density modelling to improve over another approach of *scaling partial link travel time in proportion with the length of the partial links*

2.4.2 Compressive Sensing Approach

This algorithm exploits the idea that the method of *principal component analysis* discovers the hidden structures in a large dataset of probe data [5]. Now, compressive sensing [37] is used to exploit these internal structures for performing traffic state estimation. This algorithm outperforms KNN and MSSA.

2.4.3 Curve-Fitting based Method and Vehicle-tracking based Method

These algorithms have been simultaneously proposed in [8]. In this paper [8], a method to construct an exact GIS-T digital map has also been proposed. While comparing these two algorithms vehicle tracking method was found to provide higher accuracy but at same time it takes more than double the tie taken by curve-fitting method.

2.4.4 Artificial Neural Network

In [7], an artificial neural network has been proposed comprising of 3 layers taking the vehicle's position, link id, time stamp and speed as the input to the network to calculate the link travel time. Experimental results showed that it performs better than Hellinga's model owing to the higher number of parameters used [32] [33]. Mean absolute error was found to be less than 6 % and is inversely proportional to the traffic demand. However, to train the model real GPS dataset was not used.

2.5 Comparison and Analysis of Results of Different Approaches

Table 2.1 shows the complete summary of results achieved by different techniques applied for traffic state estimation. In total 7 different techniques are compared on the basis of 7 different parameters. The important parameters are explained as follows.

i. Accuracy: It is defined as

$$\text{Ac} = 1 - \text{MAPE} \tag{2.1}$$

Where MAPE is mean absolute percentage error and is represented as

$$\frac{1}{N}\sum_{n=1}^{N}\frac{|\overline{ve}-\overline{va}|}{\overline{va}} \tag{2.2}$$

Here, N is the total number of estimated values during the experiment, $\overline{ve}$ is the estimated value of the mean speed and $\overline{va}$ is the actual value of the mean speed.

ii. Running Time: It is the time taken by the algorithm for one cycle of estimation. A cycle is the interval of time for which a new traffic state is estimated each time it gets elapsed (time granularity). For e.g. a new estimation can be made after each interval of 4 minutes or it may be an interval of 15 minutes etc.

iii. Temporal Integrity: This parameter defines the percentage of the total number of time intervals for which GPS sample points appeared for each link of the network on an average. It is worth mentioning here that the actual temporal integrity can vary from as low as ‘ 5%’ for some of the links to as high as ‘90%’ for the other links.

iv. Sampling Interval: It is defined as the time elapsed between two consecutive probe reports. A probe report is the speed and location update sent by the probe vehicle to the monitoring center.

v. Time Granularity: The traffic state from the collected data is time when a particular interval of time is elapsed. This time interval is known as the time granularity. For e.g. state can be estimated after each interval of 5 minutes or after each interval of 15 minutes.

2.6 Sources for GPS vehicle data used in the surveyed techniques

i. On 8 Feb, 2008, an experiment was conducted collectively by NOKIA, Caltrans and UC Berkeley’s department of civil and environmental engineering with the aim of collecting the traffic data by using GPS-equipped cell phone NOKIA N95 carried by the drivers of hundred vehicles while driving on a 10-mile long highway I-880 in San Francisco from 9:30 am to 6:30 pm. The data is available at [44].

ii. The [45] dataset contains mobility traces of taxi cabs in San Francisco, USA which contains GPS traces of approximately 500 taxis collected in a period of 30 days in San Francisco.

iii. This [46] dataset contains mobility traces of taxi cabs in Rome, Italy. It contains GPS traces of approximately 320 taxis collected in a period of 30 days.

Table 2.1 Comparison of techniques used for traffic state estimation

Method	Accuracy (%)	Running time (sec)	# links, # nodes	#Probe vehicles	Temporal integrity (%)	Sampling interval	Time granularity (min)
Curve-fitting	75.42	34	15000, 17000	16000	68	NA	4
Vehicle tracking	83.08	71	15000, 17000	16000	68	NA	4
Compressive sensing	75	0.827	221, NA	4000	60	30 sec – several minutes	15
Naïve KNN	55	<0.1	221, NA	4000	60	30 sec – several minutes	15
Correlation-based KNN	40	<0.1	221, NA	40000	60	30 sec – several minutes	15
Bayesian	85-90	NA	815, 527	500	NA	1 min	15
ANN	95	NA	NA	Simulated data used	NA	1 min	NA

2.7 Challenges in ITS

Accurate and reliable probe data information is very necessary for an Intelligent Transportation System to work properly. But when it comes to a large road network, it is a challenging task to monitor the traffic. As already stated in this work, large cost of most existing technologies like video cameras, roadside sensors etc. along with their limited coverage poses difficulties in getting complete traffic data.

But due to the presence of probe vehicles nowadays, the situation has been changing widely. A probe vehicle records the traffic information such as position, time stamp and speed after fixed intervals of time (e.g. 30s, 60s) along the way travelled. In contrast to the fixed sensors, the probe vehicles can provide data over any part of the road network where they are running. Probe vehicle data is being researched upon for approximately last 20 years. Some key properties of probe vehicle data are:

i. Penetration

It refers to the ratio of probe vehicles to the total number of vehicles present on the road network. Although probe vehicles like taxis etc. forms a very small part of the total number of vehicles (less than 1%) but in today's era of smartphones, a large number of vehicles can act as a probe vehicle thus enriching the traffic data.

ii. Sampling frequency

The rate at which the probe vehicles send the probe (speed, location, position) is often very low (around 1/min). Also the data is very sparse and random. This poses a challenge for accurate estimation of traffic state variables like average speed, travel time etc.

iii. Accuracy

The US GPS system provide a worst case pseudorange accuracy of 7.8 meters at a 95% confidence level (pseudo range is the distance from a GPS satellite to a receiver.) [47]. Various other factors like atmospheric effects, sky blockage and receiver quality also effects the actual accuracy attained by the users. Real world data from Federal Aviation Administration (FAA) shows that their high quality GPS receivers provide better than 3.5 meter horizontal accuracy [39] as shown by the Figure 2.1 and Figure 2.2 from the referred report.

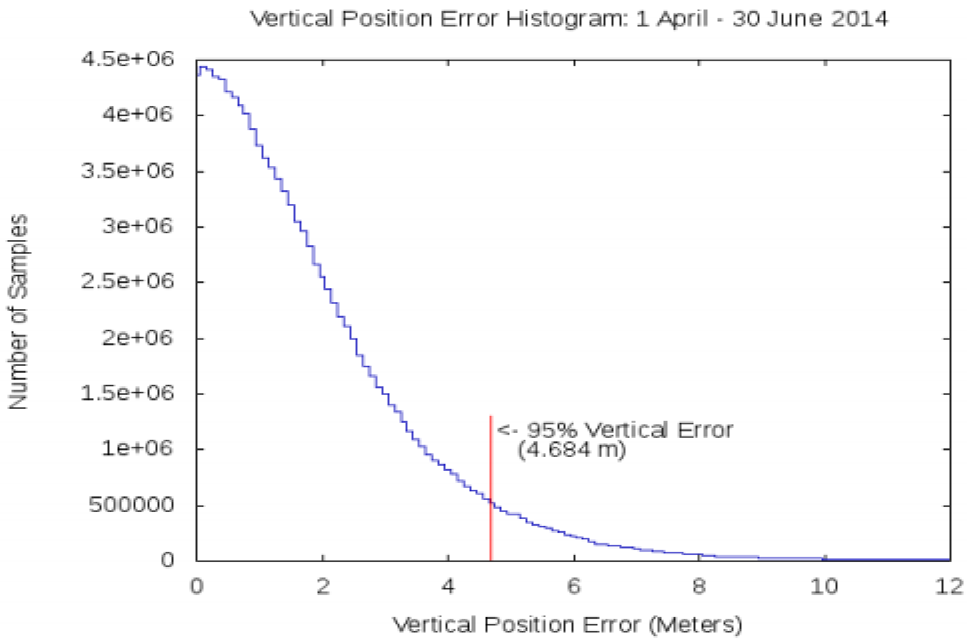

Figure 2.1 Global Vertical Error Histogram [39]

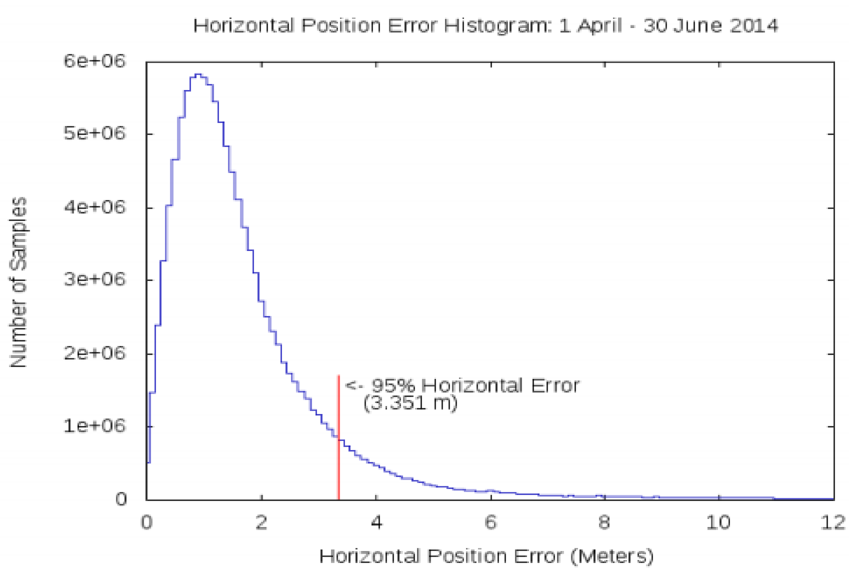

Figure 2.2 Global Horizontal Error Histogram [39]

2.8 Existing ITS Systems

Mobile century [10] was an experiment conducted in California to demonstrate the feasibility of a traffic monitoring system that GPS-enabled mobile phones. In this experiment the concept of Virtual Trip Lines [11] was used as the sampling strategy to collect the probe updates. The experiment included 100 vehicles that ran on a 10-mile highway in California and NOKIA N95 was used as the GPS device. VTL generated travel times were compared with travel times generated by loop detectors and it was suggested that "VTL measurements are more likely to be closer to the actual velocity observed on the field" [7].

A demonstration project [12] in 2011 was also started by Hitachi in Hanoi, Vietnam that collected and processed probe data collected with the help of 300 vehicles in the first year, 2011 and 800 vehicles in the second year, 2012. Graphics, data and other forms of information was the output that could be used for the estimation of traffic state. The project delivered an accuracy of approximately 70%. Another ITS project by Hitachi in Bali, Indonesia used 300 vehicles operated by a local taxi company for the calculation of travel time for a choice of route [13].

Since the past decade the potential of smartphones has been tested by researchers for carrying out many traffic related tasks such as road incident detection, traffic crowd sourcing, traffic queue length detection etc. A comprehensive survey of all the endeavors that has been done in this area has been provided by [16]. After an analysis of the existing systems that exclusively use mobile phones, [16] states that a vehicle monitoring system based on smartphones can provide an adequate performance especially for a developing country.

2.9 Conclusion

This chapter described the basic functioning of a GPS-based ITS. Primarily it focused on the recently proposed data processing techniques for measuring the traffic state by defining and comparing them on the basis of several important parameters like accuracy, sampling frequency of the vehicles and time granularity etc. In the next chapter, the problem of traffic state estimation is formally stated with objectives of this work and the methodology adopted.

Chapter 3
Problem Analysis

The previous chapter reviewed the literature that has been explored for the purpose of traffic state estimation. Various techniques have been proposed by different researchers in this area. This chapter throws light on some of the gaps that have been identified. Along with that it discusses the problem statement and finally based on the gaps that have been identified, objectives of this study have been stated.

3.1 Gap Analysis

Various traffic state estimation techniques like Bayesian, correlation based KNN, and vehicle tracking method etc. are able to predict the traffic state with accuracy less than 90%. Although Artificial Neural Network based on simulated data is able to predict with an accuracy of 95%, but it still doesn't consider the effects of events like rain, incidents etc. So, there is need to develop a model that can give better accuracy while at the same time consider the environmental factors just mentioned. Table 3.1 gives a quick review of the techniques mentioned in the literature survey.

Table 3.1 Ongoing Traffic State Prediction Techniques

Model	Bayesian	Curve-fitting	Vehicle tracking	Compressive Sensing
Accuracy (%)	85-90	75.42	83.08	75

3.2 Problem Formulation

There is a signalized road segment of some length 'L', also known as target link for which it is required to find the mean speed. Vehicles from upstream travel through this link goes to the downstream link under the control of traffic signal as shown in the Figure 3.1 [7]. When the signal is red, vehicles stop at the stop line and forms a queue, similarly when the signal is green, vehicles move out of the queue. The distribution of the probe vehicles among the traffic is random. Thus a vehicle stochastically report the probe data that includes their position, time stamp and speed. A probe point can be denoted by 3-D coordinate (t, p, s), where ‘t’ is the time stamp, ‘p’ is the GPS position and ‘s’ is the speed.

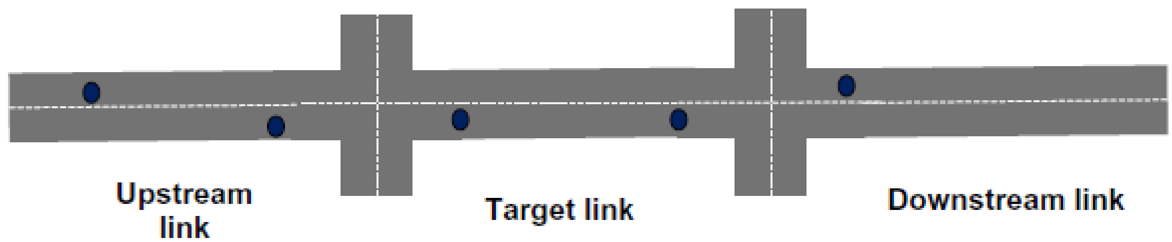

Figure 3.1 Vehicle movement [7]

The objective of this study is to estimate the mean speed of a particular link on an urban road.

3.3 Objectives

The primary goal of this research is to propose a machine learning ensemble model that can estimate the current traffic state of a link in an urban Indian road network. To achieve this aim, following objectives have been proposed.

i. To analyze and compare the latest approaches that have been proposed for the processing of GPS probe data collected with the help of probe vehicles.

ii. To develop a suitable machine learning ensemble model to measure the current traffic state of the simulated road network.

iii. To validate the developed model with simulated traffic data of an Indian city.

3.4 Methodology

Following methodology has been adopted to accomplish the above stated objectives.

i. Get acquainted with the xml representation of road maps.

ii. Understanding the simulation process of SUMO [25] traffic simulation software.

iii. Reviewing the performances of ongoing techniques for traffic prediction.

iv. Setting up the SUMO environment for simulation and traffic data generation.

v. Proposing a new model for traffic prediction.

vi. Testing the proposed model on the data generated by SUMO simulation.

3.5 Conclusion

This chapter discussed the research gaps, formally stated the problem and described the methodology to achieve the stated objectives. In the next chapter, the model for traffic state prediction has been proposed.

Chapter 4

Proposed Traffic Prediction Model

This chapter highlights the machine learning model that has been used to predict the traffic state. Figure 4.1 depicts the training of the model with generated Probe Vehicle data after which the model is used to predict the traffic state results based on testing data. The road network considered in this work comes from the urban city New Delhi, India which faces an acute traffic congestion problem. Specifically, the road link considered for estimation of average speed is Pusa road which is nearly 3 km in length with peak hour speed as 10-20 km/h.

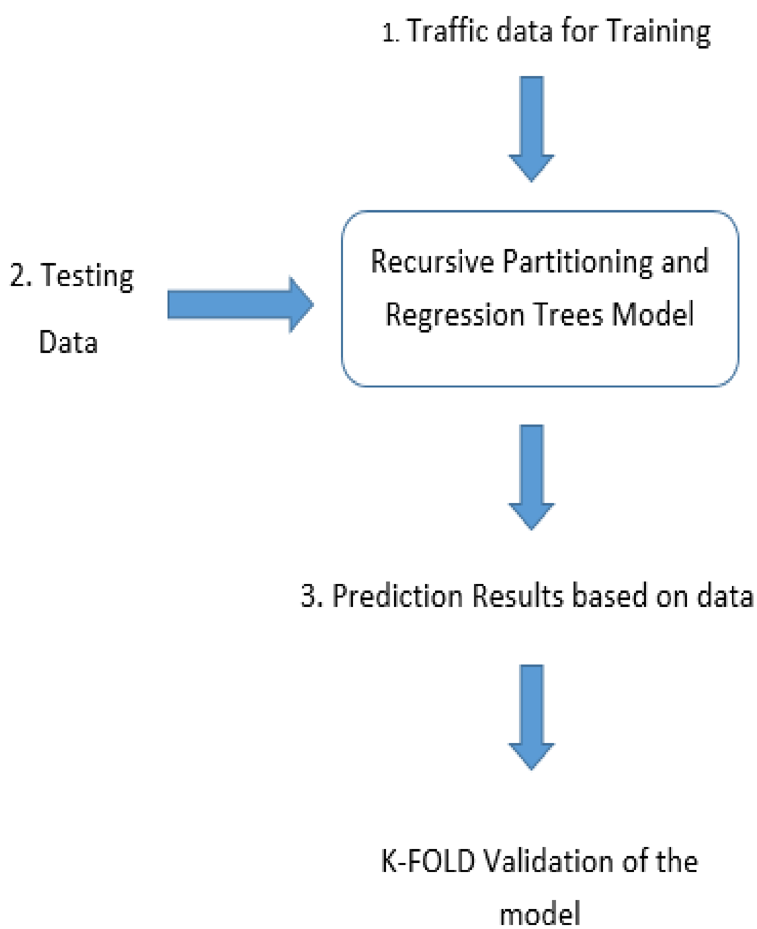

Figure 4.1 Proposed Model

4.1 Proposed Model

As shown in the Figure 4.1, the machine learning model that has been used traffic state prediction is built using "Recursive Partitioning and Regression Trees" (rpart) [40] machine learning model.

The rpart programs build classification or regression models of a very general structure using a two stage procedure; the resulting models can be represented as binary trees. The tree is built by the following process: first the single variable is found which best splits the data into two groups (`best' will be defined later). The data is separated, and then this process is applied separately to each sub-group, and so on recursively until the subgroups either reach a minimum size (5 for this data) or until no improvement can be made. The second stage of the procedure consists of using cross-validation to trim back the full tree. The details of the model can be found in [40].

To test the robustness of the proposed model, k-fold validation of the model is done. The model is available in the package 'rpart' [41] for using it in RStudio (Environment for R language implementation) or can be used as in-built functionality in Rattle (GUI for data mining R language).

4.2 Tools and Technologies

The following tools and technologies have been used in this work.

i. OpenStreetMap (OSM)

OpenStreetMap [42] is an open source project that provides free editable maps for any region of the world. To download osm map file of our desired location, openstreetmap.org is visited. The file is in format *'.osm.xml'.* This file acts as an input to the eWorld map enrichment tool. Icon of openstreetmap [43] is shown in Figure 4.1.

Figure 4.2 OpenStreetMap Icon [43]

ii. eWorld

eWorld [15] is an open source tool which can import osm files as its input and allows to emulate other factors that control the traffic such as weather effects (fog, rain etc.), incidents, maximum speed of the vehicles, traffic lights and work in progress etc. In this study, only maximum speeds of the roads has been taken into consideration. Figure 4.3 [15] shows its logo.

Figure 4.3 eWorld Icon [15]

iii. Simulation of Urban MObility (SUMO)

SUMO is an open source traffic simulation suite to model the traffic for a given map of any region. It allows to model road vehicles, pedestrians and public transport [25]. It allows to remotely control a simulation through its various APIs and can also be enhanced with customized models. Some of the supporting tools that comes along with SUMO are listed below.

- Netconvert

Netconvert.py is a tool in SUMO that converts a given osm file to generate a network file (an xm file) in a format as required by the sumo for simulation.

- Duarouter

Duarouter.py is a tool in sumo that generates vehicle routes using shortest path computation. It takes road network file as generated by netconvert.py and its primary output is rou.xml file. SUMO is available at [22].

iv. RStudio

RStudio is an open source professional software integrated development environment for R language [24]. A large number of packages that implements machine learning algorithms are available for either in-built or downloadable. Figure 4.4 shows a snapshot of RStudio GUI.

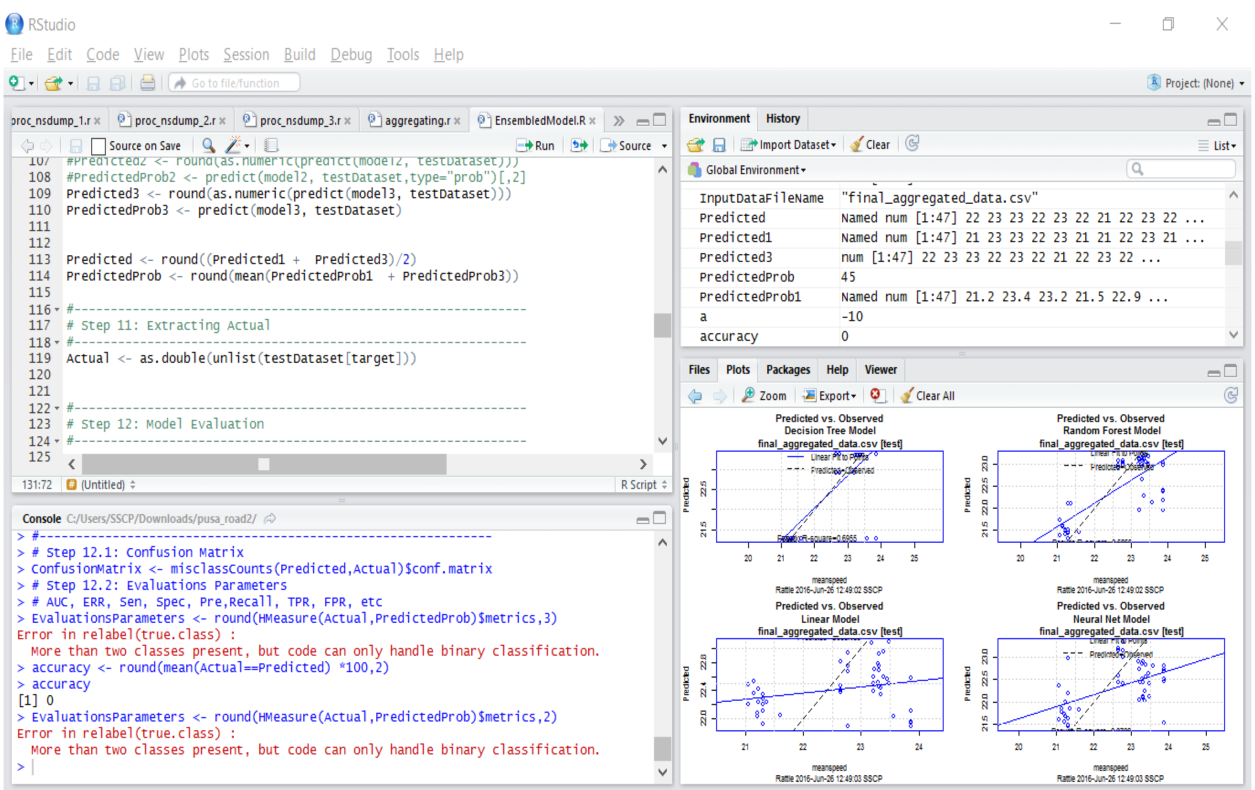

Figure 4.4 RStudio GUI

v. Rattle

Rattle is an open source GUI for data mining in R language [23]. In built packages in Rattle include randomForest, Neural Network, ada, kernlab and several others. If some package is not installed but if asked for some functionality provided by that package, Rattle gives a message that package to be installed. All the packages are available at Comprehensive R Archive Network (CRAN). Figure 4.5 shows a snapshot of Rattle GUI.

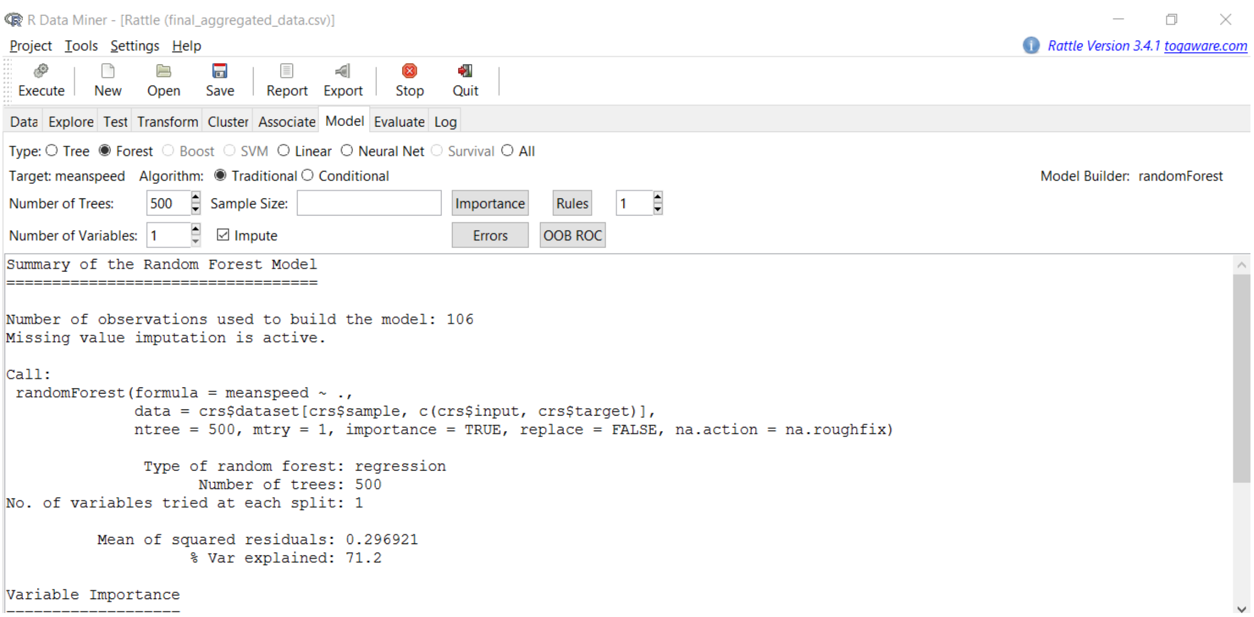

Figure 4.5 Rattle GUI

4.3 Conclusion

This chapter highlighted the model that has been proposed for traffic state prediction based on ensembling of two other models to produce better prediction results than those which are used to produce this model. Also, the tools and technologies that have been used for implementation of the proposed model and data preparation have been elaborated. In the next chapter, the experiments and results based on the things that have been stated in this chapter are discussed.

Chapter 5

Experiments and Results

This chapter discusses the implementation of operational steps that have been carried out to achieve our primary goal of traffic state prediction and the results that have been achieved. Figure 5.1 gives an overview of the complete sequence of operations after which detailed explanation of each step has been provided.

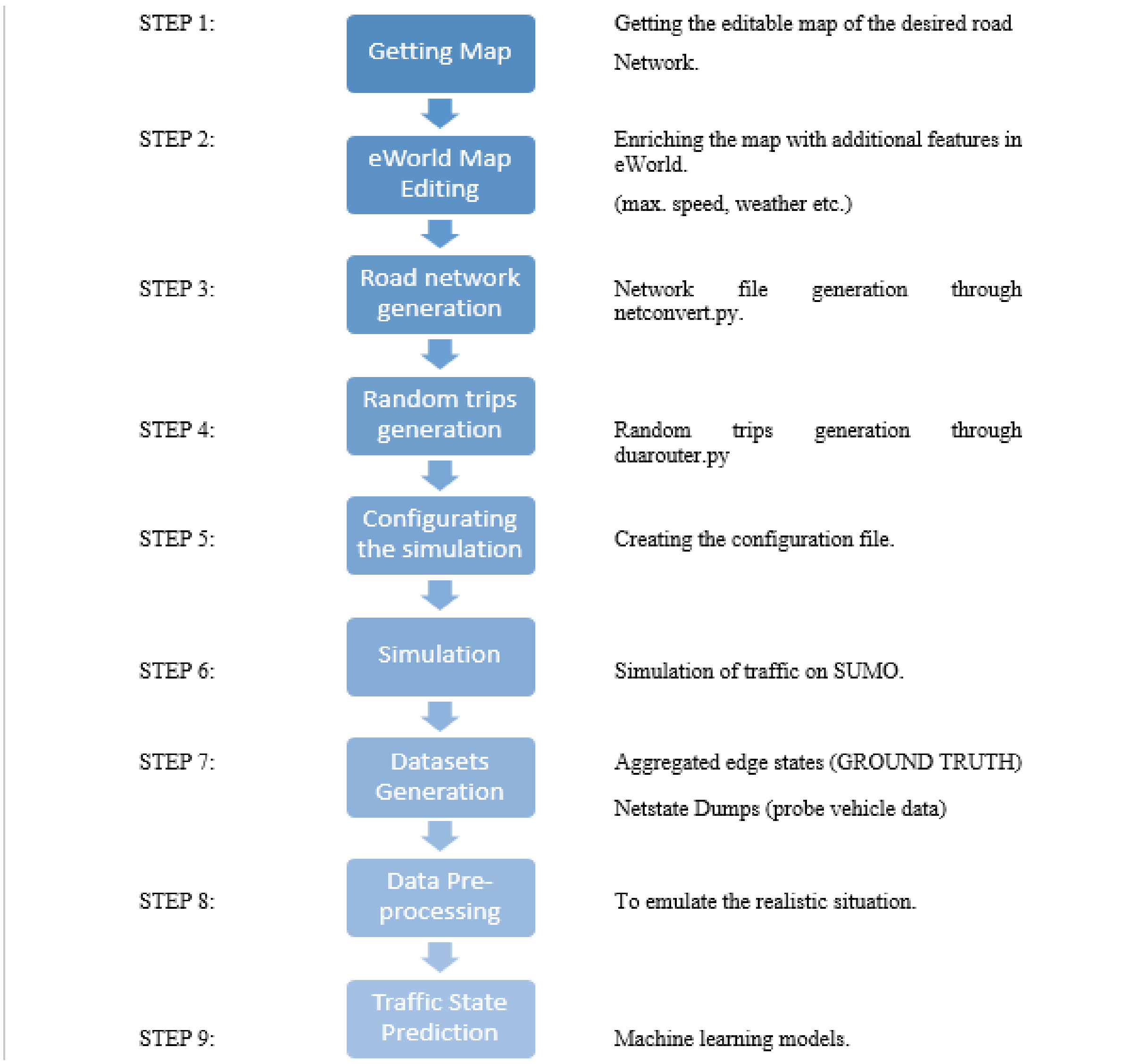

Figure 5.1 Methodological Steps for Traffic State Prediction

5.1 Implementation

Simulation of the road network to generate Probe Vehicle Data followed by traffic state estimation from the generated data goes through the following series of steps.

Step 1: Map Export

OpenStreetMap is an open source project that provides editable maps for the whole world. Exporting osm map file of New Delhi, India urban road network file from openstreetmap.org. The map contains Pusa Road which is one of the busiest road in New Delhi. Figure 5.1 shows the map of the region under consideration.

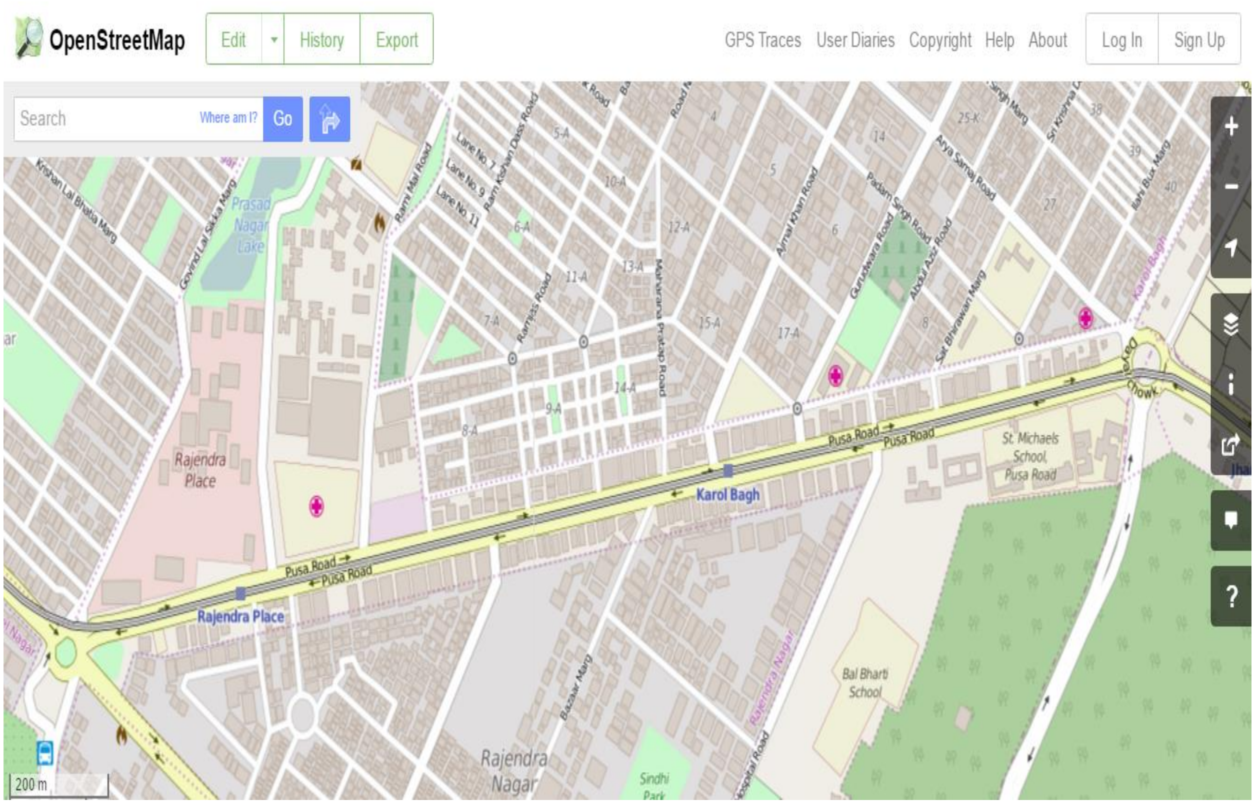

Figure 5.2 Map of New Delhi Road Network under Consideration

Step 2: Enrichment in eWorld

The .osm map file obtained in step can be further enriched in eWorld software. eWorld allows us to further enrich an .osm file with addition of new events like weather effects (fog, rain etc.), incidents, max speed, acceleration and traffic lights etc. For simplicity, here factors of max speed is taken into consideration only. Other effects like that of weather, incidents can be taken as a part of future work. Figure 5.3 shows a simplified diagram of the road network in eWorld.

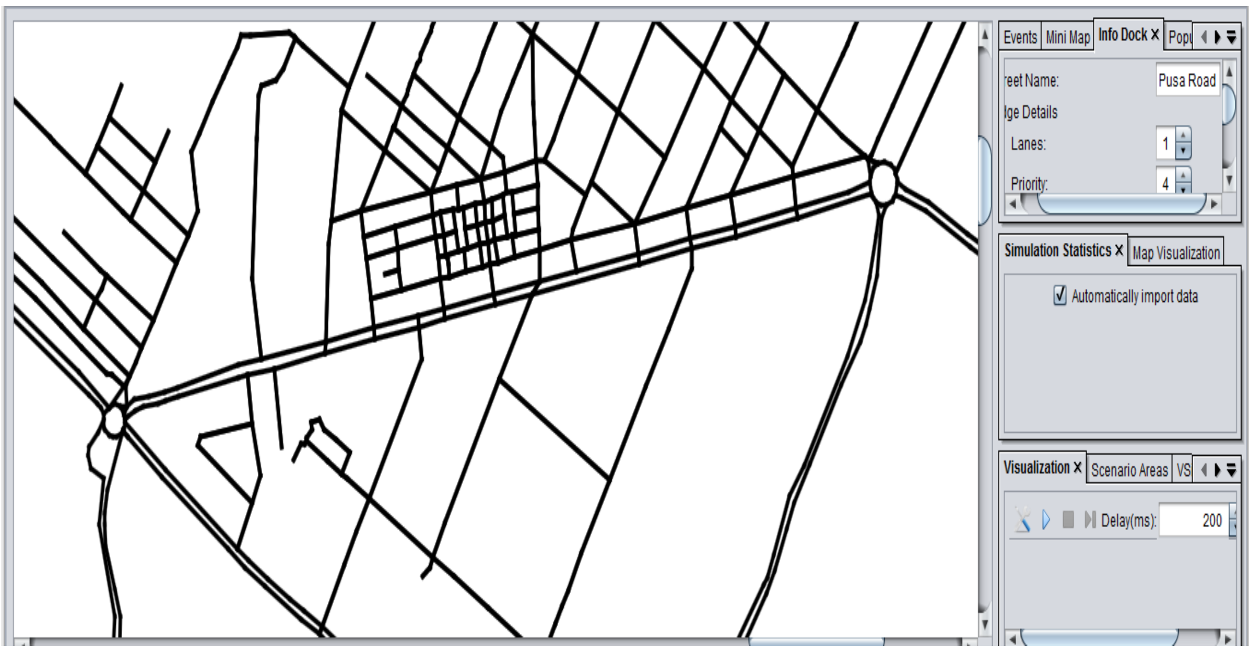

Figure 5.3 Editing of the Traffic Parameters in eWorld

Step 3: Netconvert

This tool which is a part of SUMO package take digital road networks as its input from different sources such as osm, MATsim, .net.xml, vissim etc. The exported file can be .net.xml, MATsim, .nod.xml etc. Figure 5.4 shows the output of netconverter. As it can be seen that various roads have been defined as xml tags along with their related attributes. The command used for generating the following output file *pusaroad.net.xml* from the input file *pusaroad.osm is:*

netconvert –osm-files pusaroad.osm -o pusaroad.net.xml


```xml
<net version="0.25" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://sumo.dlr.de/xsd/net_file.xs

    <location netOffset="-697680.45,-3164513.28" convBoundary="0.00,0.00,18938.16,7549.42" origBoundary="77.021810,28.592252,77.215970,28.6

    <type id="highway.bridleway" priority="1" numLanes="1" speed="2.78" allow="pedestrian" oneway="1"/>
    <type id="highway.bus_guideway" priority="1" numLanes="1" speed="8.33" allow="bus" oneway="1"/>
    <type id="highway.cycleway" priority="1" numLanes="1" speed="5.56" allow="bicycle" oneway="1"/>
    <type id="highway.footway" priority="1" numLanes="1" speed="8.33" allow="pedestrian" oneway="1"/>
    <type id="highway.ford" priority="1" numLanes="1" speed="2.78" allow="army"/>
    <type id="highway.living_street" priority="3" numLanes="1" speed="2.78" disallow="rail_urban rail rail_electric"/>
    <type id="highway.motorway" priority="13" numLanes="2" speed="44.44" disallow="rail_urban rail rail_electric moped bicycle pedestrian"
    <type id="highway.motorway_link" priority="12" numLanes="1" speed="22.22" disallow="rail_urban rail rail_electric moped bicycle pedestr
    <type id="highway.path" priority="1" numLanes="1" speed="2.78" allow="bicycle pedestrian" oneway="1"/>
    <type id="highway.pedestrian" priority="1" numLanes="1" speed="8.33" allow="pedestrian" oneway="1"/>
    <type id="highway.primary" priority="9" numLanes="2" speed="27.78" disallow="rail_urban rail rail_electric"/>
    <type id="highway.primary_link" priority="8" numLanes="1" speed="22.22" disallow="rail_urban rail rail_electric"/>
    <type id="highway.raceway" priority="14" numLanes="2" speed="83.33" allow="vip"/>
    <type id="highway.residential" priority="4" numLanes="1" speed="13.89" disallow="rail_urban rail rail_electric"/>
    <type id="highway.secondary" priority="7" numLanes="2" speed="27.78" disallow="rail_urban rail rail_electric"/>
    <type id="highway.secondary_link" priority="6" numLanes="1" speed="22.22" disallow="rail_urban rail rail_electric"/>
    <type id="highway.service" priority="2" numLanes="1" speed="5.56" allow="delivery bicycle pedestrian"/>
    <type id="highway.services" priority="1" numLanes="1" speed="8.33" disallow="rail_urban rail rail_electric"/>
    <type id="highway.stairs" priority="1" numLanes="1" speed="1.39" allow="pedestrian" oneway="1"/>
    <type id="highway.step" priority="1" numLanes="1" speed="1.39" allow="pedestrian" oneway="1"/>
    <type id="highway.steps" priority="1" numLanes="1" speed="1.39" allow="pedestrian" oneway="1"/>
    <type id="highway.tertiary" priority="6" numLanes="1" speed="22.22" disallow="rail_urban rail rail_electric"/>
    <type id="highway.tertiary_link" priority="5" numLanes="1" speed="22.22" disallow="rail_urban rail rail_electric"/>
    <type id="highway.track" priority="1" numLanes="1" speed="5.56" disallow="rail_urban rail rail_electric"/>
    <type id="highway.trunk" priority="11" numLanes="2" speed="27.78" disallow="rail_urban rail rail_electric bicycle pedestrian"/>
    <type id="highway.trunk_link" priority="10" numLanes="1" speed="22.22" disallow="rail_urban rail rail_electric bicycle pedestrian"/>
```


Figure 5.4 Road Network Generated by Netconvert

Step 4: DUAROUTER

This SUMO package tool is used to generate random routes for the vehicles on the previously generated network file. The command used is:

Python randomTrips.py –n pusaroad.net.xml –r map.rou.xml –e 3600 -l

Here 3600 defines the total simulation time of 3600 seconds and an output file named Trips.trips.py is generated. Figure 5.5 shows a snapshot of this output file. Here xml tags are defining the different trips that will be taken by the vehicles during the simulation of the vehicle.

Step 5: Creation of sumo configuration file

A sumo configuration file has to be created in which the use of the above generated network file, route file and some other additional files are defined. Figure 5.6 shows the contents of the configuration file.

```xml
<?xml version="1.0"?>
<!-- generated on 2016-06-18 15:48:37.939000 by $Id: randomTrips.py 18096 2015-03-17 09:50:59Z behrisch $
  options: -n pusaroad.net.xml -r pusaroad.rou.xml <doubleminus>intermediate 400 -l
-->
<trips>
    <trip id="0" depart="0.00" from="80932759#0" to="31058500#9" via="80932789#1 231369828 265927790 80932742#13 181810384#0 -28190265#2 28190305#12 80932742#13 80932742#10 28
    <trip id="1" depart="1.00" from="31058500#4" to="28190305#4" via="31058500#12 423861855 80932778#3 28190296#2 231369829#0 80932788#2 31058500#4 28190296#2 -28190265#1 259
    <trip id="2" depart="2.00" from="259087606#0" to="-28190265#1" via="80932750#0 77852919#1 80932759#0 265927771#2 231369837#1 80932757#1 31058500#6 28190430#6 235103523 24
    <trip id="3" depart="3.00" from="77852919#1" to="31058500#1" via="231414760#1 28190305#6 -28190265#1 31058500#3 80932789#5 -28190265#2 46481776#1 77852919#1 348492664#2 2
    <trip id="4" depart="4.00" from="24676573#4" to="176466568#0" via="80932786#7 424013690 259087606#3 80932750#11 265927770#2 31058500#4 28190305#7 231369843 28190305#1 2351
    <trip id="5" depart="5.00" from="31058500#12" to="176466568#0" via="80932742#7 31058500#7 80932789#4 265927771#0 176466568#0 -28190265#1 80932788#2 80932742#12 80932750#0
    <trip id="6" depart="6.00" from="80932778#0" to="77852919#1" via="28190296#1 231414760#0 46481776#2 348492664#3 28190296#2 28190305#9 80936829#0 28190430#2 231414760#1 -28
    <trip id="7" depart="7.00" from="259081663#2" to="80932788#5" via="181810386 24676573#2 31058500#4 31058500#11 28190395#5 231414760#0 265927790 293429094#3 80932750#9 281
    <trip id="8" depart="8.00" from="293429094#3" to="80932789#6" via="31058500#5 31058500#6 259087606#1 235103423#1 231369843 265927790 80932742#13 259081663#0 80932742#3 -28
    <trip id="9" depart="9.00" from="80932735#3" to="28190430#10" via="231369842 265927770#1 -28190265#1 259083225#0 31058500#6 80932786#9 231369831#1 235103523 265927790 809
    <trip id="10" depart="10.00" from="80932773#7" to="28190430#5" via="176466568#0 231369828 28190305#8 31058500#1 80932757#3 28190430#7 348492664#1 28190430#5 265927771#2 8
    <trip id="11" depart="11.00" from="259081663#9" to="28190305#2" via="235103423#1 259081663#0 80932742#14 80932772#2 24676122#1 231369843 31058500#0 80932763#0 80932789#1
    <trip id="12" depart="12.00" from="293429094#3" to="265927770#1" via="80932737#1 24676573#4 80932780#1 231369843 24676573#4 80932750#10 265927771#1 80932796#2 31058500#0
    <trip id="13" depart="13.00" from="231369842" to="265927790" via="80932757#3 80932742#12 265927778#1 46481776#2 348492664#0 231369843 80932763#0 28190296#2 31058500#9 281
    <trip id="14" depart="14.00" from="28190296#3" to="24676573#3" via="176466568#0 181810384#2 265927771#0 80932789#4 24676573#2 80932742#14 80932757#1 80932789#2 176466568#
    <trip id="15" depart="15.00" from="80932742#10" to="176466568#0" via="80932742#13 80932750#12 28190296#1 80936829#0 80932757#1 31058500#9 28190334#3 28190430#7 231369828
    <trip id="16" depart="16.00" from="265927771#1" to="77852919#1" via="28190296#3 231369828 176466568#0 80932759#0 231369828 265927778#1 24676573#5 -28190265#2 -28190265#1
    <trip id="17" depart="17.00" from="231369828" to="181810384#5" via="31058500#4 80932786#9 31058500#12 28190296#3 80932750#4 80932767#1 176466568#0 80932773#1 31058500#12
    <trip id="18" depart="18.00" from="80932757#1" to="80932796#3" via="80932773#3 259081663#1 259083238#0 80932788#1 80932742#14 28190296#3 77852919#1 24676573#4 423861855 8
    <trip id="19" depart="19.00" from="265927790" to="31058500#1" via="24676573#5 77852919#1 80932757#1 28190430#3 80932773#3 28190305#13 28190395#5 80932788#6 31058500#1 231
    <trip id="20" depart="20.00" from="231414760#0" to="80932788#5" via="31058500#7 348492664#2 28190430#8 176466568#0 28190305#6 231369843 46481776#2 24676573#2 -28190265#2
    <trip id="21" depart="21.00" from="265927771#2" to="46481776#2" via="265927771#2 28190296#3 80932782#0 80932758#1 31058500#11 80932788#0 80932759#1 28190305#13 80932788#2
    <trip id="22" depart="22.00" from="80932759#0" to="265927771#3" via="176466568#0 265927770#0 231369843 80932742#13 265927790 28190305#2 24676533#1 80932789#1 77852919#1 2
    <trip id="23" depart="23.00" from="80932773#1" to="80932763#1" via="176466568#0 80932735#2 80932742#11 24676573#5 80932750#0 -28190265#1 -28190265#2 80932773#3 176466568#
```

Figure 5.5 Trips generated by DUAROUTER

```xml
<?xml version="1.0" encoding="iso-8859-1"?>

<configuration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://sumo.sf.net/xsd/sumoConfiguration.xsd">

    <input>
        <net-file value="pusaroad.net.xml"/>
        <route-files value="pusaroad.rou.xml"/>
        <additional-files value="pusaroad.poly.xml"/>
    </input>
    <time>
    <begin value="0"/>
    <end value="3600"/>
    </time>
    <report>
        <no-duration-log value="true"/>
        <no-step-log value="true"/>
    </report>
    <output>
  <netstate-dump value="netstate_dump.xml" />
</output>


</configuration>
```

Figure 5.6 Sumo configuration file

Step 6: Simulation Run

Now after the configuration file is generated, the simulation can be run on sumo with the following command.

Sumo-gui pusaroad.sumo.cfg

Figure 5.7 shows a snapshot of running vehicles on the simulated network that includes one of the busiest roads (pusa road) of New Delhi, India.

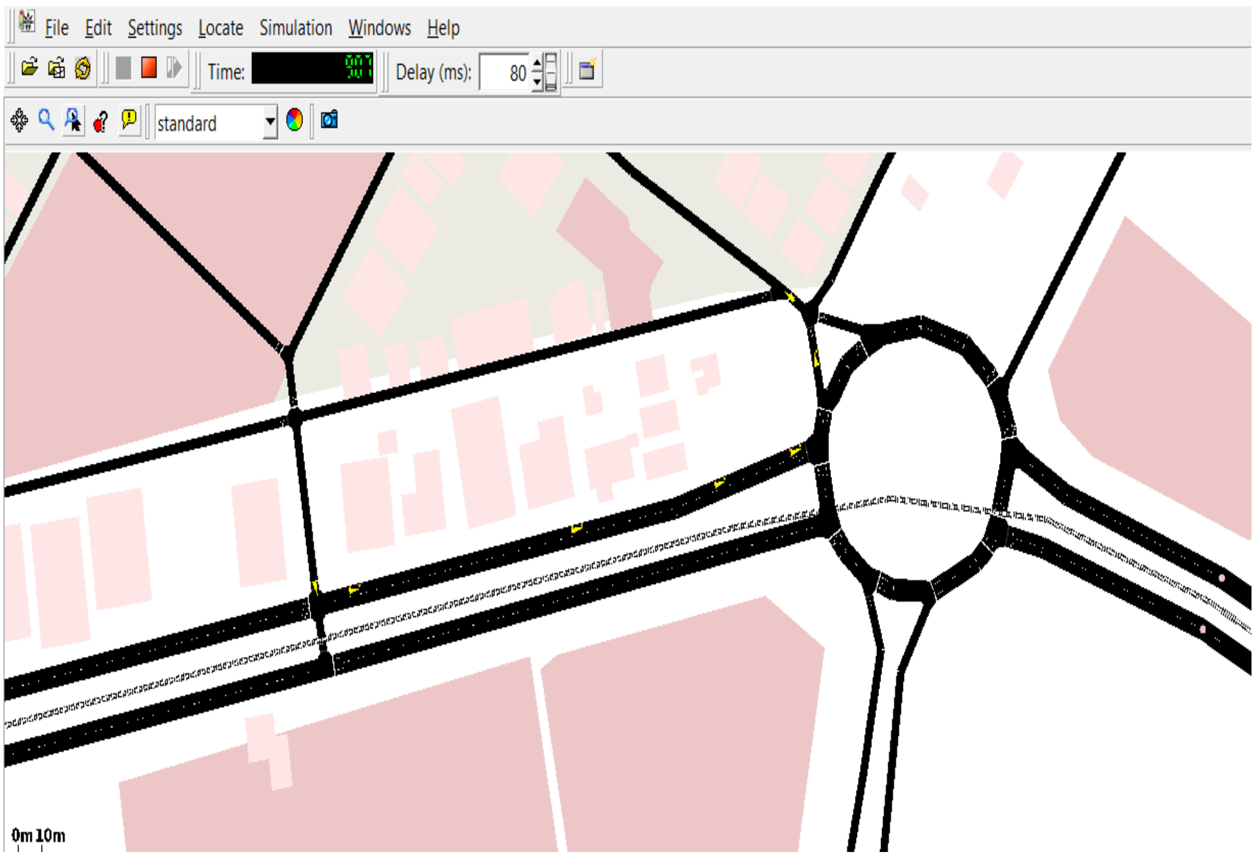

Figure 5.7 Snapshot of Traffic Simulation at a Particular Instant

Step 7: Generation of Probe Vehicle Data (PVD)

With the help of the configuration file, PVD of all the vehicles (speed, position, id, time, type etc.) running on the network is generated with the help of following command (Figure 5.8) and gets written in the file netstatedump.xml. Along with, ground truth data about edge states (mean speed, travel time) is generated with the help of option "--additional-files" option. Figure 5.9 shows a snapshot of netstatedump file.

```
C:\Users\SSCP\Downloads\pusa_road2>sumo -c pusaroad.sumo.cfg --random --netstate
-dump netstatedump.xml -- additional-files additional.xml
```

Figure 5.8 PVD generation

```
<netstate xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://sumo.dlr.de/xsd/netstate_file.xsd">
    <timestep time="0.00"/>
    <timestep time="1.00">
        <edge id="31058500#4">
            <lane id="31058500#4_0">
                <vehicle id="1" pos="5.10" speed="0.00"/>
            </lane>
            <lane id="31058500#4_1"/>
        </edge>
    </timestep>
    <timestep time="2.00">
        <edge id="31058500#4">
            <lane id="31058500#4_0">
                <vehicle id="1" pos="6.54" speed="1.44"/>
            </lane>
            <lane id="31058500#4_1"/>
        </edge>
    </timestep>
    <timestep time="3.00">
        <edge id="31058500#4">
            <lane id="31058500#4_0">
                <vehicle id="1" pos="9.66" speed="3.12"/>
            </lane>
            <lane id="31058500#4_1"/>
        </edge>
        <edge id="77852919#1">
            <lane id="77852919#1_0">
                <vehicle id="3" pos="5.10" speed="0.00"/>
            </lane>
            <lane id="77852919#1_1"/>
        </edge>
```

Figure 5.9 Probe Vehicle Data

Step 8: Data Pre-Processing

Floating car data generated in the above step comes from each and every vehicle running on the road for every second (as can be seen in the Figure 5.9). However, in real situation, probe vehicles like taxis, buses etc. constitute only up to 5% of the total traffic from which the probe data (penetration rate) is collected. Also, the data is delivered at some sampling interval like 60 sec, 30 sec etc. But before this, above generated xml files are converted into csv files for further processing. Figure 5.10 shows the process of data pre-processing. As a result of this pre-processing the final data generated is shown in Figure 5.11.

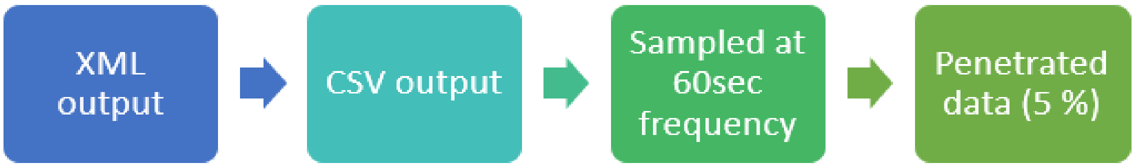

Figure 5.10 Steps involved in data Pre-processing.

	A	B	C	D	E	F
1	timestep_time	vehicle_speed	vehicle_id	vehicle_pos	meanspeed	
2	60	23.86	38	133.18	22.77	
3	120	12.11	5	22.27	23.78	
4	120	8.16	22	352.26	22.65	
5	120	16.19	46	80.31	22.65	
6	120	26.72	33	169.53	23.86	
7	180	16.48	97	70.75	22.65	
8	240	15.12	97	338.5	22.65	
9	240	27.51	187	129.21	23.86	
10	300	27.26	5	276.99	23.78	

Figure 5.11 Final Pre-processed Data

Step 9: Model Training and Prediction

Now the model is trained with the pre-processed data in Rattle (GUI for data mining with R language). Figure 5.12 shows the snapshot of Rattle just before the training when the input attributes are defined to be the time step, vehicle position and vehicle speed while the corresponding output is set to be the mean speed of the road. When the model is executed on the data provided, a comma separated values (csv) file is generated for the predicted values as shown in the Figure 5.11. A graph showing all the actual values with corresponding predicted values is shown in the Figure 5.13, where R^2 is the coefficient of determination ('R' is the correlation value.).

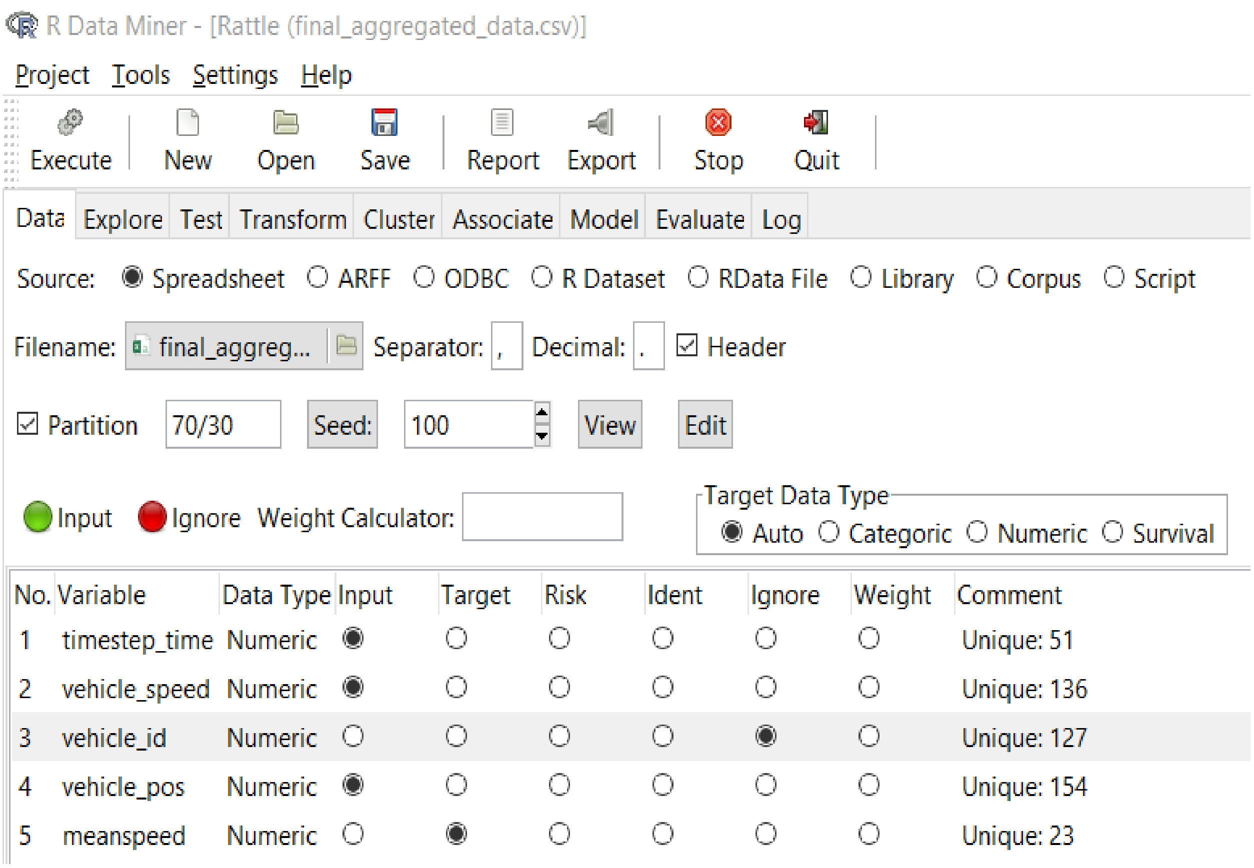

Figure 5.12 Setting of input and output attributes in Rattle

	A	B	C
1	meanspeed	rpart	
2	23.78	23.32894737	
3	23.86	23.32894737	
4	22.65	23.32894737	
5	22.65	23.32894737	
6	23.57	23.32894737	
7	23.78	23.32894737	
8	23.86	23.32894737	
9	23.78	23.32894737	
10	21.06	23.32894737	

Figure 5.13 Actual mean speed (m/sec) and Predicted speed values

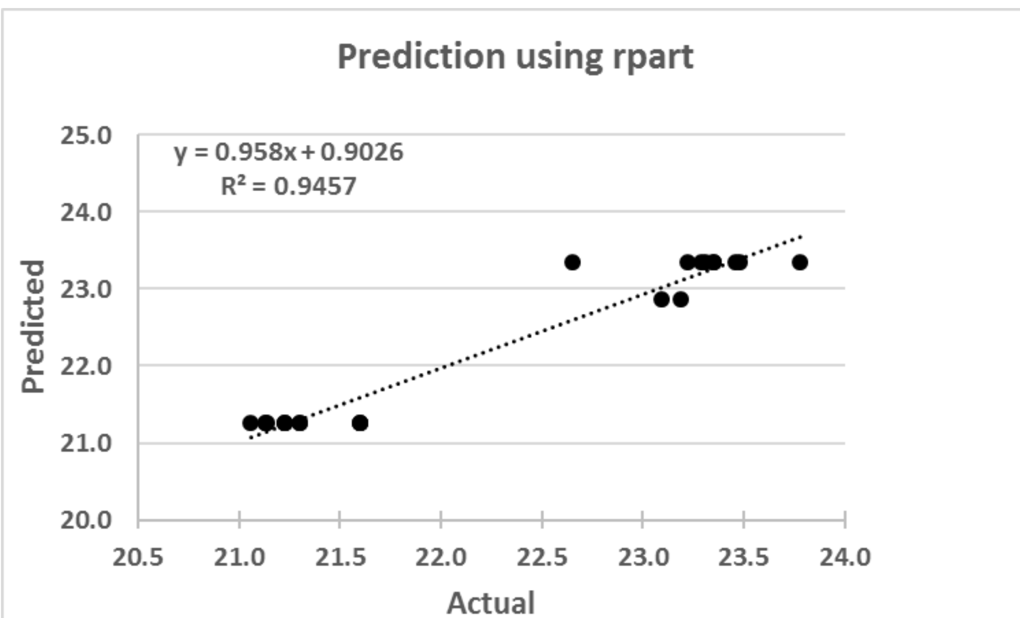

Figure 5.14 Actual mean speed (m/sec) vs Predicted mean speed values

5.2 Results

To find out average accuracy of the proposed model, K-fold validation of the model is done i.e. the model is run 10 times (here K=10) (to test the robustness of our model) as shown in the Figure 5.15. The results after the validation are presented below.

5.2.1 Coefficient of Determination (R^2)

This value describes how good the predicted values are replication of the actual values [14]. If the regression model comes out to be perfect then the value of R^2 is 1 and if the regression model completely fails to replicate the actual values then the value of R^2 is 0.

The average value of R^2 after K-fold validation for the proposed model is 0.87 which indicates a good replication of the actual values.

5.2.2 Accuracy

Accuracy measures how often the model makes correct prediction with an acceptable error (in case of regression). It is calculated as:

$$\text{Accuracy} = \frac{100}{n}\sum_{i=1}^{n} q_i \tag{5.1}$$

$$q_i = \begin{cases} 1, & if\ abs(p_i - a_i) \leq err \\ 0, & otherwise \end{cases} \tag{5.2}$$

where ‘a’ is the actual value, ‘p’ is the predicted value, *err* is the acceptable error and ‘n’ is the total number of instances. Figure 5.14 shows a graph between the actual values and the predicted values.

After K-fold validation, the average accuracy of the proposed model is 98.48% with an acceptable error of +1 or -1.

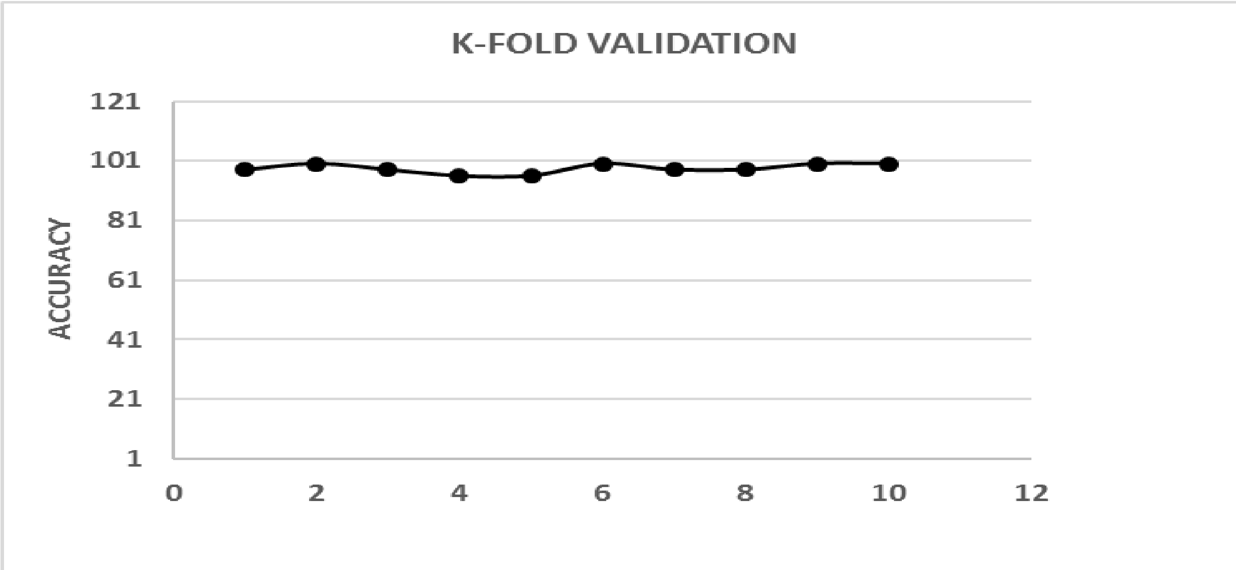

Figure 5.15 K-Fold Validation

5.2.3 Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error is defined as:

$$\text{MAPE} = 100 * \frac{1}{N}\sum_{n=1}^{N}\frac{|\overline{ve}-\overline{va}|}{\overline{va}} \qquad (5.3)$$

Here, N is the total number of estimated values during the experiment, $\overline{ve}$ is the estimated value of the mean speed and $\overline{va}$ is the actual value of the mean speed. Average MAPE value for the proposed model comes out to be 1.148% whereas Artificial Neural Network Model proposed in [7] achieved MAPE of 3.97%. This model took travel time as a metric for traffic state estimation whereas in this work average speed is chosen for representing the traffic state. So, MAPE values indicate better results for the proposed model. Table 5.1 shows a comparison of different parameters of performance measurement between ANN model and the model proposed in this work. Figure 5.16 shows a graph to compare the MAPE values of the two models.

The reason for choosing ANN for comparison with the proposed model (RPART) is that [7] also used simulated data rather than actual data for training and validation of ANN. Also, among all the models compared in literature survey, ANN has the highest accuracy among all of them.

Table 5.1 Performance Comparison of ANN and RPART for Traffic State Prediction

Model Name	**Accuracy (%)**	**MAPE (%)**
ANN	95	3.97
RPART (Proposed Model)	98.48	1.148

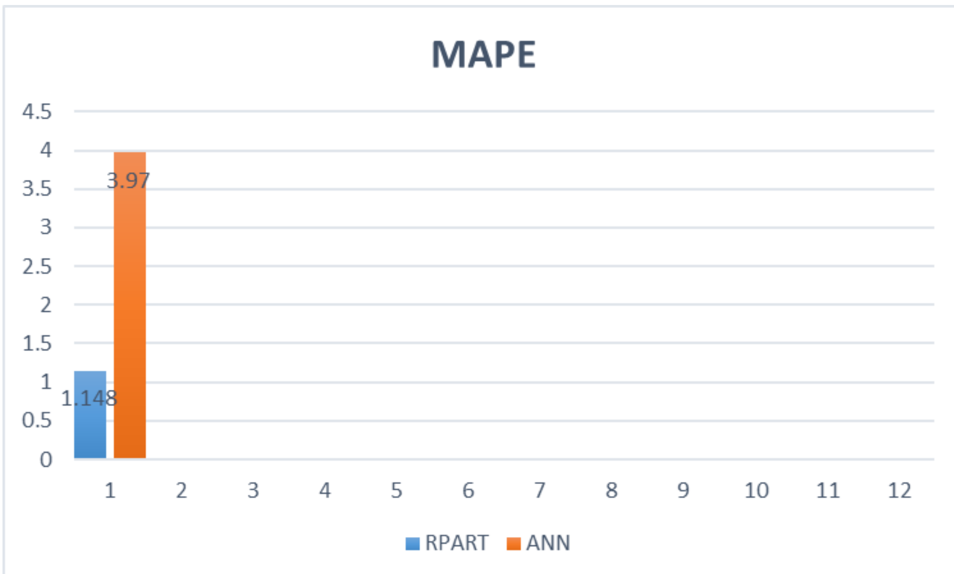

Figure 5.16 Mean Absolute Percentage Error for ANN and RPART

5.3 Conclusion

This chapter discussed the implementation details of the work carried out to achieve the primary objective of traffic state prediction along with the results produced by the proposed model which suggests that the proposed model is a robust model with an average accuracy of 98.3% with an acceptable error of +1 or -1 and also a comparison between ANN and the proposed RPART model has been done. The next chapter summarizes up the whole work and throws light on some future work that can be accomplished.

Chapter 6
Conclusions and Future Work

This chapter concludes the whole work presented in this thesis and also provides some future work that can be done to take the work at a higher level.

6.1 Conclusions

In this work, Intelligent Transportation Systems (ITS) based on GPS based data gathering technique has been discussed. Primarily, it focused on traffic data processing techniques which are used for traffic state estimation in the form of average speed of the vehicles or the average travel time. This state estimation provides the traffic authorities to manage the traffic on the available road infrastructure efficiently and also helps in planning of urban infrastructure.

Challenges of GPS based ITS systems relating to the quality of data (sparse and uneven distribution) have also been discussed and finally a machine learning model has been proposed for efficient traffic state estimation from sparse probe data which is validated using simulated data.

6.2 Thesis Contribution

- Comprehensive discussion and comparison of traffic data processing techniques like Bayesian, Compressive Sensing and ANN etc. have been discussed.
- Traffic dataset is generated based on an Indian city road network using SUMO simulation software.
- Recursive Partitioning and Regression Trees (RPART) model is applied on the generated dataset to measure traffic state.
- The proposed model (RPART) is compared with a previously proposed ANN model on the basis of mean absolute error (MAPE) which shows better accuracy of the RPART model.

6.3 Future Work

- The proposed model can be tested for real GPS data.
- Other effects such as rain, fog and incident etc. can also be taken into consideration.
- The work can be used to run a cloud based application that delivers traffic state to its users.
- The model can be further be implemented for a large dataset of traffic information.

References

[1] [Online]. Available: http://www.technobuffalo.com [Accessed 26 5 2016]

[2] [Online]. Available: http://www.business2community.com [Accessed 25 5 2016]

[3] [Online]. Available: http://www.thoughtsonCloud.com [Accessed 25 5 2016]

[4] Hofleitner, A., Herring, R., Abbeel, P., & Bayen, A.:" Learning the dynamics of arterial traffic from probe data using a dynamic Bayesian network. Intelligent Transportation Systems", *IEEE Transactions on Intelligent Transportation Systems* vol. 13, no. 4, pp. 1679--1693, 2012.

[5] Zhu, Y., Li, Z., Zhu, H., Li, M., & Zhang, Q.: "A compressive sensing approach to urban traffic estimation with probe vehicles", *IEEE Transactions on Mobile Computing.* vol. 12, no.11, pp. 2289—2302, 2013

[6] Jenelius, E., & Koutsopoulos, H. N.: "Travel time estimation for urban road networks using low frequency probe vehicle data." *Transportation Research Part B: Methodological*. 53,pp. 64—81, 2013

[7] Zheng, F., & Van Zuylen, H.:" Urban link travel time estimation based on sparse probe vehicle data" *Transportation Research Part C: Emerging Technologies*. 31, pp 145—157, 2013

[8] Kong, Q. J., Zhao, Q., Wei, C., & Liu, Y.: "Efficient traffic state estimation for large-scale urban road networks", *IEEE Transactions Intelligent Transportation Systems*, vol. 14 no.1, pp. 398—407, 2013

[9] Vlahogianni, E. I., Golias, J. C., & Karlaftis, M. G.: "Short-term traffic forecasting: Overview of objectives and methods" *Transport reviews* vol. 24 no. 5, pp. 533--557 ,2004

[10] Herrera, J. C., Work, D. B., Herring, R., Ban, X. J., Jacobson, Q., & Bayen, A. M. "Evaluation of traffic data obtained via GPS-enabled mobile phones: The Mobile Century field experiment". *Transportation Research Part C: Emerging Technologies* vol. 18 no.4, pp. 568—583, 2010

[11] Hoh, B., Gruteser, M., Herring, R., Ban, J., Work, D., Herrera, J. C., & Jacobson, Q. "Virtual trip lines for distributed privacy-preserving traffic monitoring" *In Proceedings of the 6th international conference on Mobile systems, applications, and services*, pp. 15--28. ACM, June, 2008

[12] Okubo, T., Yoshioka, K., Nakamura, A., & Taniguchi, N.: "Realizing Smart Mobility Using Probe Data" *Hitachi Review* vol. 63 no.6, pp. 359, 2014

[13] Morioka, M., Kuramochi, K., Mishina, Y., Akiyama, T., & Taniguchi, N.: "City Management Platform Using Big Data from People and Traffic Flows", *Hitachi Review*. vol. 64 no.1, pp 53, 2015

[14] [Online]. Available: https://en.wikipedia.org/wiki/Coefficient_of_determination [Accessed 20 6 2016]

[15] eWorld homepage [Online]. Available: http://eworld.sourceforge.net/ [Accessed 20 4 16]

[16] Engelbrecht, J., Booysen, M. J., van Rooyen, G. J., & Bruwer, F. J.: Survey of smartphone-based sensing in vehicles for Intelligent Transportation System applications. Intelligent Transport Systems, IET. 9(10), 924--935 (2015)

[17] Djahel, S., Doolan, R., Muntean, G. M., & Murphy, J.: "A communications-oriented perspective on traffic management systems for smart cities: challenges and innovative approaches", *IEEE Communications Surveys & Tutorials*. vol. 17 no. 1, pp 125—151, 2015

[18] Bramberger, M., Brunner, J., Rinner, B., & Schwabach, H.: "Real-time video analysis on an embedded smart camera for traffic surveillance". *In Real-Time and Embedded Technology and Applications Symposium, 2004. Proceedings. RTAS 2004. 10th IEEE*, pp. 174--181, May, 2004,

[19] Coifman, B.: "Estimating travel times and vehicle trajectories on freeways using dual loop detectors". *Transportation Research Part A: Policy and Practice.* vol. 36 no. 4, pp 351—364, 2002

[20] Kong, Q. J., Li, Z., Chen, Y., & Liu, Y.: "An approach to urban traffic state estimation by fusing multisource information", *IEEE Transactions on Intelligent Transportation Systems*,. vol. 10 no. 3, pp 499—511, 2009

[21] [Online]. Available: http://www.technobuffalo.com [Accessed 15 2 2016]

[22] [Online]. Available: http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000 [Accessed 10 1 2016]

[23] Williams, G. J. Rattle: a data mining GUI for R. The R Journal, vol. 1 no. 2,pp 45-55, 2009

[24] Racine, J. S. "RStudio: A Platform-Independent IDE for R and Sweave. Journal of Applied Econometrics", vol. 27 no.1, pp 167-172, 2012

[25] Behrisch, M., Bieker, L., Erdmann, J., & Krajzewicz, D. “SUMO–simulation of urban mobility: an overview.” *In Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation, ThinkMind*, 2011

[26] Talia, D. “Toward cloud-based big-data analytics.” *IEEE Computer Science,* pp 98-101, 2013

[27] Jadeja, Y., & Modi, K. Cloud computing-concepts, architecture and challenges. In Computing, Electronics and Electrical Technologies (ICCEET), IEEE International Conference on pp 877-880 , March, 2012

[28] Qureshi, K. N., & Abdullah, A. H. (2013). “A survey on Intelligent Transportation Systems” *Middle-East Journal of Scientific Research*, vol. 15 no.5, pp 629--642.

[29] CAO, P. “Utilization of Probe Vehicle Data to Estimate Urban Traffic Conditions”, 2014

[30] Caceres, N., Romero, L. M., Benitez, F. G., & del Castillo, J. M. “Traffic flow estimation models using cellular phone data” *IEEE Transactions on Intelligent Transportation Systems*, vol. 13 no. 3, pp 1430--1441, 2012

[31] Hiribarren, G., & Herrera, J. C. (2014). “Real time traffic states estimation on arterials based on trajectory data” *Transportation Research Part B: Methodological*, vol. 69, pp 19--30.

[32] Hellinga, B., Izadpanah, P., Takada, H., & Fu, L. (2008). “Decomposing travel times measured by probe-based traffic monitoring systems to individual road segments.” *Transportation Research Part C: Emerging Technologies*, vol. 16 no. 6, pp 768--782.

[33] Hellinga, B. R., & Fu, L. “Reducing bias in probe-based arterial link travel time estimates.” *Transportation Research Part C: Emerging Technologies*, vol. 10 no. 4, pp 257—273, 2002

[34] Jenelius, E., & Koutsopoulos, H. N. (2013). “Travel time estimation for urban road networks using low frequency probe vehicle data.” *Transportation Research Part B: Methodological*, vol. 53, pp 64--81.

[35] Leduc, G. “Road traffic data: Collection methods and applications. Working Papers on Energy,” *Transport and Climate Change*, vol. 1 no.55, 2008

[36] Zou, L., Xu, J. M., & Zhu, L. X. “Arterial speed studies with taxi equipped with global positioning receivers as probe vehicle.” *In Proceedings. IEEE International Conference on Wireless Communications, Networking and Mobile Computing*, 2005, vol. 2, pp. 1343-1347, September 2005

[37] Donoho, D. L. (2006). “Compressed sensing”. *IEEE Transactions on information theory,* vol. 52 no. 4, pp. 1289--1306.

[38] [Online]. Available: http://hybridcloudsolutions.info/are-you-moving-to-cloud-what-are-the-benefits-you-get-from-cloud-computing [Accessed 10 6 2016]

[39] [Online]. Available: http://www.nstb.tc.faa.gov/reports/PAN86_0714.pdf#page=22 [Accessed 12 6 2016]

[40] Therneau, Terry M., and Elizabeth J. Atkinson. "An introduction to recursive partitioning using the RPART routines.", 1997

[41] [Online]. Available: https://cran.r-project.org/web/packages/rpart/rpart.pdf [Accessed 17 6 16]

[42] Haklay, M., & Weber, P. (2008). “Openstreetmap: User-generated street maps.” *IEEE Pervasive Computing,* vol. 7 no. 4, pp. 12--18.

[43] [Online]. Available: https://commons.wikimedia.org/wiki/File:Openstreetmap_logo.svg [Accessed 15 4 16]

[44] [Online]. Available: http://traffic.berkeley.edu/project/mobilecentury [Accessed 21 1 16]

[45] [Online]. http://crawdad.org/epfl/mobility/20090224/ [Accessed 21 1 16]

[46] [Online]. http://crawdad.org/news.html [Accessed 21 1 16]

[47] [Online]. http://www.gps.gov/systems/gps/performance/accuracy [Accessed 6 6 16]

List of Publications

Accepted

V. Mehta and I. Chana, “Urban Traffic State Estimation Techniques Using Probe Vehicles: A Review”, in *International Research Symposium on Computing and Network Sustainability*, *(IRSCNS 2016)*, Goa, 1-2 July, 2016**. [to be published in Springer LNNS series]**

Video Link

https://www.youtube.com/channel/UCVk5kb64rVcFd-TeY2HTgGg