People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
University Of Larbi Tebessi –Tebessa

Faculty Of Exact Sciences And Sciences Of Nature and life
Department Of Mathematics And computer science

Final thesis
For obtaining the MASTER diploma
Field : Mathematics And Computer Sciences
speciality : Computer Sciences
Option : Networks And IT Security

Theme

Learning Dynamics on Vehicular Networks

Presented By :
Ziadeddine Makhlouf

Before the jury:

Mr A.Zeggari	MCB	University Of Larbi Tebessi	President
Mr A.Gahmous	MAA	University Of Larbi Tebessi	Examiner
Mr M.Gasmi	MCB	University Of Larbi Tebessi	Supervisor
Mr A.Sahraoui	MCB	University Of Larbi Tebessi	Supervisor

Abstract

Traffic congestion is one of the biggest challenges facing modern services, it costs us time, money and even human lives in some extreme cases. In order to solve or rather prevent this problem, researchers must predict its state, and among all the techniques used we have chosen deep learning because studies show its superiority over all traditional methods in most fields. In our thesis, we understand the basics of car traffic and delve into its concepts, characteristics, challenges and methods used to predict its state, to this end, we offer a deep learning architecture. The main problem with the vehicular traffic is its unpredictable variables which are in most cases caused by human behavior and weather changes, so we need to include these factors in our research. In this context, the traffic state of vehicles is defined as the situation of vehicles on the road network, this state could be: free movement, congestion or a state between them.

Based on a review of the literature on traffic forecasts, a sequential deep learning architecture was proposed, this model was formed by data sets extracted from a simulated scenario. The results indicate a great capacity for learning the proposed model with low loss values. In our work, we tried to explore different aspects of the problem of traffic congestion, and explored the solutions proposed by the researcher, we found that this field is very large, its problems evolve over time and become more difficult simultaneously with the development of the proposed solutions and their effectiveness.

Keywords: Traffic Flow, Vehicular Ad-hoc Network, Machines Learning, Deep Learning, Traffic Simulation, Prediction Techniques

Résumé

La congestion du trafic est l'un des plus grands défis auxquels sont confrontées les sociétés modernes, il nous coûte du temps, de l'argent et même des vies humaines dans certains cas extrêmes. Afin de résoudre ou plutôt de prévenir ce problème, les chercheurs doivent prévoir son état, et parmi toutes les techniques utilisées nous avons choisi l'apprentissage profond car les études montrent sa supériorité sur toutes les méthodes traditionnelles dans la plupart des domaines. Dans notre thèse, nous comprenons les bases de la circulation automobile et plongeons dans ses notions, caractéristiques, défis et méthodes utilisées pour prévoir son état, à cette fin, nous proposons une architecture d'apprentissage en profondeur. Le principal problème de la circulation automobile est ses variables imprévisibles qui sont dans la plupart des cas causées par les comportements humains et les changements météorologiques, nous devons donc inclure ces facteurs dans nos recherches. Dans ce contexte, l'état de circulation des véhicules est défini comme la situation des véhicules sur le réseau routier, cet état pourrait être: la libre circulation, la congestion ou un état entre eux.

Sur la base d'une revue de la littérature sur les prévisions de trafic, une architecture séquentielle d'apprentissage en profondeur a été proposée, ce modèle a été formé par données extraites d'un scénario simulé. Les résultats indiquent une grande capacité d'apprentissage du modèle proposé avec de faibles valeurs de perte. Dans notre travail, nous avons essayé d'explorer différents aspects du problème de la congestion du trafic, et exploré les solutions proposées par les chercheurs, nous avons constaté que ce domaine est très vaste, ses problèmes évoluent avec le temps et deviennent plus difficiles simultanément avec le développement de les solutions proposées et leur efficacité.

Mots Clés : Flux de Trafic, VANet, Apprentissage Automatique, Apprentissage Profond, Simulation de trafic, Techniques de Prédiction

ملخص

يعد الازدحام المروري أحد أكبر التحديات التي تواجه المجتمعات الحديثة ، ويكلفنا الوقت والمال وحتى حياة البشر في بعض الحالات القصوى. من أجل حل هذه المشكلة أو بالأحرى منعها ، يجب على الباحثين توقع حالتها ، ومن بين جميع التقنيات المستخدمة ، اخترنا التعلم العميق لأن الدراسات تظهر تفوقها فوق جميع الأساليب التقليدية في معظم المجالات. في أطروحتنا نحن نفهم أساسيات حركة مرور المركبات ونغوص في مفاهيمها وخصائصها وتحدياتها والمعتقدات المستخدمة للتنبؤ بحالتها ، لهذا الغرض نقترح بنية تعلم عميقة. المشكلة الرئيسية لحركة مرور المركبات هي متغيراتها التي لا يمكن التنبؤ بها والتي تحدث في معظم الحالات بسبب السلوكيات البشرية والتغيرات المناخية ، لذلك يجب علينا تضمين هذه العوامل في بحثنا. في هذا السياق ، يتم تعريف حالة مرور المركبات على أنها حالة المركبات في شبكة الطرق ، ويمكن أن تكون هذه الحالة: التدفق الحر أو الازدحام أو حالة بينهما.

استنادًا إلى مراجعة الأدبيات حول التنبؤ بالمرور ، تم اقتراح هيكل متعمق للتعلم العميق ، تم تدريب هذا النموذج من خلال مجموعات البيانات المستخرجة من سيناريو محاكاة. تشير النتائج إلى قدرات تعليمية عالية للنموذج المقترح مع قيم خسارة منخفضة. في عملنا ، حاولنا استكشاف جوانب مختلفة لمشكلة الازدحام المروري ، واستكشفنا الحلول التي اقترحها الباحث ، وجدنا أن هذا المجال كبير جدًا ، ومشاكله تتغير بمرور الوقت وتصبح أكثر تحديًا في نفس الوقت مع تطور الحلول المقترحة وكفاءته.

Dedication

To my dear parents, for all their sacrifices, their love, their tenderness, their support and their prayers throughout my studies,

To my dear Sister Mayar for its permanent encouragement, and its moral support,

To my dear brother Ihab, for its support and encouragement,

*To all my friends, especially "Adnene, Mohame,*Yaakoub *and Naoual" for their support throughout my university career,*

May this work be the fulfillment of your alleged vows, and the leak of your infallible support,

Thanks for always being there for me.

Appreciation

By way of recognition, I would like to express my sincere thanks to all the people who contributed directly or indirectly to the smooth running of my end-of-study thesis and to the development of this modest work.

My sincere gratitude to Mr.Mohamed Gasmi and Mr.Abdellatif Sahraoui for the quality of their teaching, their advices and their undeniable interest that they bring to all students.

In the impossibility of mentioning all the names, our sincere thanks go to all those who, directly or indirectly, have allowed by their advice and their skills the realization of this thesis

Finally, I would not forget to thank the entire faculty of for the enormous work they do to create the most favorable conditions for the conduct of our studies.

General Introduction

We're living in a world that everything happens rapidly, and every aspect of our lives is changing and becoming more suitable to this word, technological development entered every field and made our daily tasks easier, but in the other hand it opened a myriad number of challenges and horizons, we are no longer facing the previous century issues, our nowadays challenges are more sophisticated, complex and were considered unattainable some decades ago.

In its journey to build smart cities, humanity have come a long way, and this can be seen in the widespread of use of information and communication technologies, our cities today can make

Intelligent responses to different kinds of needs. Among the various notable goals of smart cities, building intelligent traffic systems with its main component, vehicular urban traffic.

Vehicular traffic is of great importance in modern societies, because of its contribution in the development of countries' economies and fighting against poverty by providing access to employment, social, health and education services, and with the opportunities it gives us as individuals in our daily lives, it also opens more areas and stimulate economic and social development. Its quality became an indicator of countries' levels of development.

Despite the advancement of vehicular traffic there are still many problems that are continuously growing and have to be solved, these problems vary from air pollution, accidents to fuel consumption but the major issue is congestion, which became an inescapable condition in big metropolitan areas especially during peak-hours. There have been many attempts to to prevent or rather cope with traffic congestion, and the most promising solution is simply to predict it, which is the approach that we will discuss in our thesis.

Predicting traffic means that drivers will know where and when congestion will occur, this idea was studied and developed in the last decades, and many techniques were created to increase the accuracy of prediction, these techniques started from basic statistical methods and turned to more complex mathematical models with more parameters taken in consideration. In recent years, the emergence of the digital era resulted in a massive data explosion across the globe, and as a result of these developments, "big data" could now be "mined" from various open channels such as roadside detectors, car navigation systems and GPS-equipped smartphones, this data can be collected and analyzed, we need discover and recognize the patterns and regularities around this data, which is the area where artificial intelligence excels.

Deep learning is the most used technique in traffic flow prediction; it’s a machine learning technique that teaches computers to do something that humans do naturally, Studies show that deep learning completely surpasses traditional methods in most of areas. The most important advantage of deep learning is replacing handcrafted features with efficient algorithms for unsupervised or semi-supervised feature learning and hierarchical feature extraction. Using Neural Network for modeling traffic flow and congestion prediction came to the picture in 1993 in[1], and in [2] the difference between deep learning techniques and statistical techniques was shown. The data regarding Traffic Flow and Traffic Congestion are two instances of Spatio-temporal data. They embady a location (Spatial Feature) and a time (Temporal feature), which create a non-linearity in data, that issue was addressed[3]. Because of thenature of traffic data as time series, many author applied Feed Forward Neural Networks (FFNNs) with a backpropagation algorithm[4], after that researcher implemented CNN[5], RNN[6] and LSTM[7].

Our thesis will be divided in three chapters, the first is an introduction to vehicular networks and traffic dynamics, in which we will dive into traffic flow notions, we will define traffic flow characteristics, challenges and how to manage it, after that we will its components and how to collect and measure them, in addition to their types, next we will talk about some vehicular traffic applications and performance indicators, at the end of the first chapter we will discuss traffic flow theories. The second chapter will be more about the problematic of our thesis which is the implementation of deep learning in traffic flow forecasting, with explanations of some used prediction techniques. The last chapter will be about our practical work, in which we will explain our chosen model and simulated scenario, at the end of this chapter there will the results of our applied model.

Summary

Chapter 02 Deep Learning Architecture For Traffic Flow Prediction

Chapter 03 CNN Deep Larning Architecture Based On Traffic Simulation Model For Traffic Flow Pediction

Figure Li

Chapter 01

Introduction to Vehicular Networks and Traffic Dynamics

1. The Vehicular Traffic
2. Traffic data collection
3. Traffic Application
4. Traffic Dataset Taypes
5. Traffic Data Modeling
6. Traffic Performance Indicators
7. Traffic Flow Theories

INTRODUCTION

Drivers in daily life move to large cities experiencing significant congestion in their trajectory. Indeed, traffic congestion is not a current phenomenon, where it emerges with the start of the automobile and the continuous increase vehicles on the road. Congestion remains a global phenomenon, on most points of interest such as highways and major cities. This is a very important factor affecting the economic sectors, public transport, freight transport and travelers. Its impacts are reflected in an increase in travel time. This extra time leads to additional fuel consumption, a pollen environment, undesirable social behavior, risks of accidents and incidents.

The evolution of large cities requires the deployment of Intelligent Transport Systems (ITS) in order to increase road safety. These systems promote safer driving by improving the driving experience and the travel time of cars to their points of interest.

Instead of investing in transport infrastructure, traffic flow forecasting promotes forecasting tools to optimize the use of infrastructure. Understanding the nature of traffic flow trends for a road infrastructure rather than a single road is a big challenge. The objective of traffic prediction is to improve the quality of information disseminated by traffic applications. This information becomes precise and important in helping drivers make the best choices, helping traffic managers manage a large road network and dynamically allocating vehicle resources.

In this chapter, we will highlight the phenomenon of vehicle traffic, vehicle dynamics and its recent challenges, how to manage it and what parameters can be managed. After that, we look at some vehicle applications that can increase the efficiency of the roads. Finally, we present traffic performance indicators and traffic flow theories.

1. The Vehicular Traffic

Looking back at the evolution of road safety, it is interesting to see how much has changed, most of which has happened in the past decades. From carriage to sensors, cameras and Bluetooth technology, a lot has happened in the area of road safety in recent decades. Traffic congestion, often severe enough to require drastic control measures, was at less a characteristic of urban life. from Roman times. A fundamental cause, then as today, was poor urban planning, with roads laid out in such a way as to bring traffic from all the districts to a central crossing point. In the 1st century BC, Julius Caesar banned wheeled traffic from Rome during the day, a measure gradually extended to cities in the provinces.

At the end of the 1st century AD, Emperor Hadrian was forced to limit the total number of carts entering Rome.

Around 1500 Leonardo da Vinci, envisioning a revolutionary solution to the problems of urban traffic - then real in overcrowded and busy Italian cities - proposed to separate traffic on wheels and pedestrians by creating routes at different levels. With the exception of the railroad, however, few separate route systems were established before the 20th century.

Congestion was severe enough in European cities in the 17th century to require orders prohibiting parking on certain streets and establishing one-way traffic. The advent of the railroad brought temporary relief to the growing problem of traffic control, although it created congestion in terminals inside cities. The automobile, with its growth first in speed and then in number in relation to horse-drawn transport, quickly created a new situation which would become one of the characteristic problems of urban industrialized society in the 20th century.

Today, all drivers on the road can be victims on the roads. Road safety is essential for different users on the roads to ensure smooth driving. The transport services are still trying to put in place a set of measures to prevent road users from being killed or seriously injured in road accidents, or to mitigate the consequences. The road safety field of the last decades is completed in parallel with the information technology revolution. The term "road safety" instantly conjures up images of modern cars today, road accidents occurred even before the invention of the motor vehicle. The humble horse and cart, when used as both a freight and passenger carrier, combined with a lack of traffic rules, have resulted in numerous accidents, injuries and deaths.

1.1.Vehicular Traffic Challenges

The main difference between the traffic phenomenon and other social phenomena is its inverse growth. Simply, with the improvement of social indicators such as security, well-being and the economy, urban travel increases, which increase traffic congestion. With increasing urbanization and cheaper access to vehicles, traffic congestion is a major recurring problem in many large cities. Traffic jams cost us time, money and health. In addition, vehicles in congestion can burn up to 80% more fuel than those in free traffic, which results in greater air pollution [4]. According to data published in 2010, cars emit around 5.53 million kg of carbon dioxide into the atmosphere each year, or 16% of the world total [5].

Motor vehicles also produce 72% nitrogen oxide and 52% reactive hydrocarbons worldwide [6].

These greenhouse gases, by trapping heat in the atmosphere, cause a rise in global temperatures which was 0.71 C in 2017.

A warmer global atmosphere affects agriculture, wildlife, sea level and natural landscapes. In addition, the health risks from air pollution are extremely serious. Poor air quality causes respiratory illnesses, such as asthma and bronchitis.and the risk of developing dangerous illnesses such as cancer is increased, and the healthcare system faces significant medical costs.

Airborne particles alone are responsible for around 30,000 premature deaths worldwide [7] Moscow followed with 91 hours lost due to congestion, while London and Paris lost 74 and 69 hours respectively [8]. On average, congestion costs in Los Angeles are $ 19.2 billion for the city as a whole. Costs are higher in New York, at $ 33.7 billion. This cost for all London commuters is $ 12.2 billion and for Berlin $ 7.5 billion. The economic damage caused by congestion last year in the United States, Germany and Great Britain totaled US $ 461 billion. These costs increase as the world's population and urbanization increase [9].

1.2.Vehicular Traffic Management

Transport problems plagued humans long before the advent of cars. However, in recent years, traffic jams have become particularly acute in cities around the world: too many vehicles on too few roads! Traffic jams and congested roads are a daily problem. The growing demand for mobility is also a major challenge. The increase in traffic increases safety, health, the environment and the economy. The resulting costs can be measured as differential delay, vehicle operating costs (fuel and wear), accidents, polluting emissions and driver stress. Intelligent traffic management systems can reduce congestion and associated costs by optimizing the use of transport resources and the infrastructure of the transport system as a whole, thereby bringing more efficiency in the areas of traffic flow and reliability of transport services.

Congestion-reduction measures can be thought of as falling into two categories:

- **Temporary** measures free up road capacity that is soon filled by induced demand: people adapt their lifestyles to prevailing road conditions. Such measures are therefore worth pursuing only if they either buy time or lay the foundations for more radical interventions.
- **Virtuous** measures start a feedback loop that induces more and more people to make a modal shift away from driving. Making a bus service more convenient or cheaper will

increase patronage, which means that the service can be run more frequently and for longer hours, making it convenient and attractive to more people.

1.3.Parameters Of Vehicular Traffic

Vehicular traffic is composed of many components, parameters of these components differ depending on our point of view, and we distinguish three categories of data.

1.3.1. Microscopic data

Road traffic flows are made up of drivers associated with individual vehicles, with its own characteristics. These characteristics are said to be microscopic when a traffic flow is considered to be composed of such a flow of vehicles. The dynamic aspects of these traffic flows are formed by the underlying interactions between the drivers of the vehicles. This is largely determined by the behavior of each driver, as well as by the physical characteristics of the vehicles.

Because the process of participating in a traffic flow is strongly based on the behavioral aspects associated with human drivers [10], it seems important to include these human factors in the modeling equations. However, this leads to a sharp increase in complexity, which is not always a desired artifact [11].

In the microscopic model, we always consider a vehicle-driver combination as a single entity, taking into account only certain traffic characteristics linked to the vehicle.

a. Vehicle related variables

Considering the individual vehicles, we can say that each vehicle i in a lane of a traffic flow has the following information variables :

- a length, denoted by (li)
- a longitudinal position, denoted by (xi)
- a speed, denoted by $vi = \frac{dxi}{dt}$(1)
- an acceleration, denoted by $\alpha i = \frac{dvi}{dt} = \frac{d^2 xi}{dt^2}$(2)

Note that the position x_i of a vehicle is typically taken to be the position of its rear bumper. In this first approach, a vehicle's other spatial characteristics (i.e., its width, height, and lane number) are neglected. And in spite of our narrow focus on the vehicle itself, the above list of variables is also complemented with a driver's reaction time, denoted by τ_i .

With respect to the acceleration characteristics, it should be noted that these are in fact not only dependent on the vehicle's engine, but also on e.g., the road's inclination, being a non-negligible factor that plays an important role in the forming of congestion at bridges and tunnels.

Except in the acceleration capabilities of a vehicle, we ignore the physical forces that act on a vehicle, e.g., the earth's gravitational pull, road and wind friction, centrifugal forces [12].

b. Traffic flow characteristics

Referring to Figure 1 (l), we can consider two consecutive vehicles in the same lane in a traffic stream: a follower (i) and its leader($i + l$).

From the Figure, it can be seen that vehicle (i)has a certain space headway(hsi)toits predecessor it is expressed in meters , composed of the distance called the space gap(g_{si})to this leader and its own length (li):

$$H_{si}=g_{si}+l_i(3)$$

By taking, as stated before, the rear bumper as a vehicle's position, the space headway

$$H_{si} = x_i + 1 - x_i \quad (4)$$

The space gap is thus measured from a vehicle's front bumper to its leader's rear bumper.

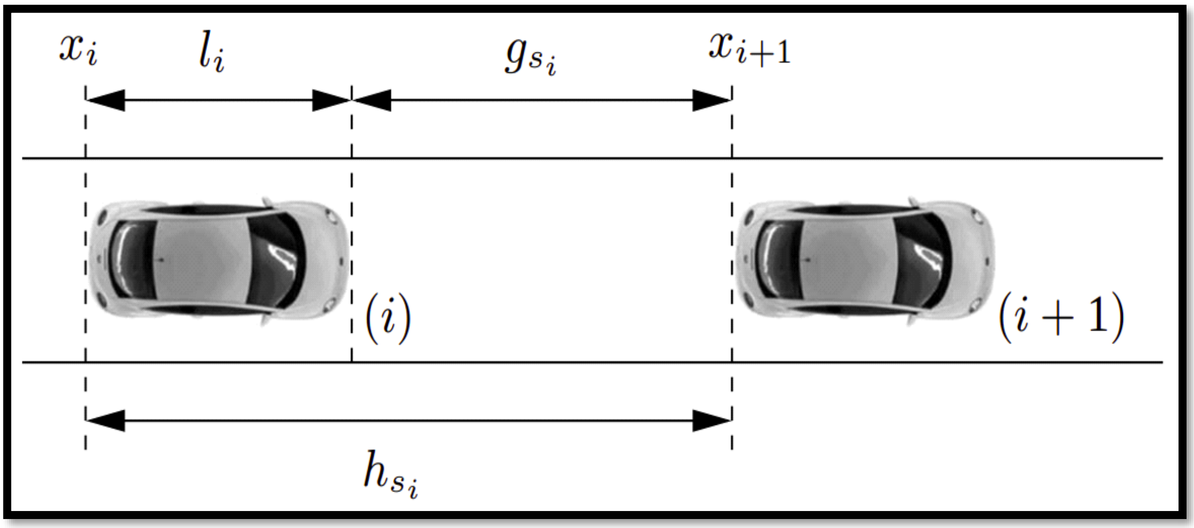


Figure 1 : Two vehicles (*i* and *i*+ *l*)[113]

1.3.2. Macroscopic Data

Macroscopic view is observing the phenomenon in a bigger picture and "zooming it out" Instead of considering each vehicle in individually, (traffic streams are regarded e.g., as a fluid).

a. Density

Density is a numeric macroscopic characteristic that means how crowded is a certain section of road, it's a macroscopic characteristic, it's expressed by the numbr of vehicle per kilometer/mile .Density only considers the abstract quantity 'number of vehicles' which means that it totally ignores the effects of traffic composition and vehicle lengths.

Because density can be measured in a spatial region and computed for temporal regions., density has to be estimated when it cannot be exactly measured or computed, or when its measurements are faulty. To this end, several available techniques exist e.g., based on explicit simulation using a traffic flow propagation model [13], based on a vehicle reidentification system [14], based on a complete traffic state estimator using an extended Kalman filter [15], or based on a non-linear adaptive observer [16], . . .

Using the spatial region R_s, the density k for single-lane traffic is defined as:

$$k = \frac{N}{K} \quad (5)$$

With N the number of vehicles present on the road segment. If we consider multi-lane traffic, we have to sum the partial densities k_l of each of the L lanes as follows:

2. Traffic Data Collection

The traffic data collection aims to capture traffic row data which accurately reflectthe traffic situation in a road section. For instance, it can be counting

the number of vehicles traveling on a road or collecting information on journey time, current speed, density, passage time, delay time, occupation, etc. Indeed, there are two approaches for collecting traffic data, namely:

2.1.Manual Traffic Data Collection

The most common methods of collecting traffic data are manual methods. These methods can be classified as manual counting and survey.

a. Manual counting

A manual counting of traffic data is carried out to determine the classification of vehicles, the percentage of turns at intersections, the movement of pedestrians or the occupation of vehicles; it is generally applied by a person using a counting board to count all vehicles at the intersection or path selected for a predetermined period of time. Data can be manually recorded by one of these three methods [20]: tally sheets, mechanical counting boards, and electronic counting boards.

- **Tally Sheets**: Data can be saved with a check sign in a pre-prepared dataset. A stopwatch is required to measure the desired time interval.
- **Mechanical Counting Boards**: Mechanical meters include meters mounted on a page that record each line of the route. Common forms of these signs include the number of pedestrians, bicycles, vehicles and the volume of traffic. Typical counters are push

buttons with three to five registers. Each button is used to count a specific type of vehicle or pedestrian. A stopwatch or stopwatch is also required to measure the desired time interval.

- **Electronic Counting Boards**: Electronic counting tables are tools used to collect traffic counting data that uses the battery. Electronic metering is more compact and easier to use than mechanical cards. They are supplied with battery and have an internal clock that automatically separates the time intervals for data collection. In addition, data can be downloaded to a computer, saving time and reducing human error.

This method of data collection can be costly in terms of labor. However, in cases where it is necessary to collect data relating to the classification of different vehicles separately or where the infrastructure of automatic methods is not used, these methods must be used. [21].

b. Survey

This method is used alongside household questionnaires to estimate the origin-destination matrix (ODM) [12], it requires experience, skills and a good understanding of the study area to obtain a correct estimate of the ODM. It is also important to know the purpose of the study and the detail of the modeling methods because the data are affected by these properties. In addition, many practical considerations, including the availability of time and money, have a great impact on the design of the survey.

2.2.Automatic Data Collection

The different types of sensors used today are different from those of the first generations of ATMS and ITS systems [22], because the data sources used were presence sensors in fixed positions such as inductive loop detectors, capable of detect the presence of nearby vehicles.

- **Inductive loop detectors** are installed in the roads for measuring the change in the magnetic field, they can detect the presence of a conductive metal object.
- **Magnetic detectors** to sense the magnetic anomaly ferrous metal objects cause in the Earth’s magnetic field.
- **Video image processing** provide traffic flow data across several lanes through analyzing the video image of roadway surveillance cameras.
- **Microwave radar sensors** transmit electromagnetic signals and receive echoes from objects of interest.
- **Infrared sensors** operate in two modes, in active mode illuminate the detection zones

with low power infrared energy transmitted by laser diodes then use the reflected energy to detect vehicles. In passive mode, these sensors detect the energy emitted by vehicles or the energy emitted by the atmosphere and reflected by vehicles.

- **Laser radar sensors** are active sensors that transmit scanning infrared beams in the near infrared spectrum over one or more lanes.
- **Audio sensors** calculate traffic density or volume by using different audio signal processing techniques.

Nowadays, traffic data can be collected from GPS systems that can be integrated in smartphones, vehicles or RSU. The provided datasets by traffic detectors presents traffic data sources with supplement presence-type sensors. These data include information about road infrastructure, smart devices, vehicles and roads, which have not been covered with presence sensors yet. The advantage of GPS sources is the historical and real-time traffic trajectories, which have allowed us to make better forecasts of traffic flows for ATMS and ITS. The use of mobile crowd sensing techniques have helped us collect the trajectories of pedestrian vehicles and cellphones [23], providing us with valuable data on the trajectories of vehicles and pedestrians.

a. Data from Fixed Position Sensors

The deployment of presence sensors / detectors at a fixed position in space (indicated by p) is the traditional approach; this approach meant that the sensors always measured at a specific point on the road (they could measure one or more lanes, depending on the capabilities of the sensor used). If we study only one direction of a road segment (for example, using an inductive loop detector), the data from a traditional fixed position sensor can be described as an ordered sequence of measurements m̄p in a position p given:

$$\boldsymbol{\overline{m}_p} = \{\boldsymbol{\overline{m}_{p,t}}\} \quad (6)$$

$$t = 1,2, \ldots T \quad (7)$$

Where $\overline{m}_{p,t}$ is the value of the measurement at time t and position p.

b. Data from Moving Sensors

As we mentioned earlier, we can collect mobile data from smartphones and GPS equipped vehicles (such as taxis or bikes), which are now considered mobile sensors, these sensors provide us with valuable data on the trajectories of vehicles and pedestrians using mobile crowd detection techniques. Many crowd-sensing applications deal with tasks related to urban transport systems, which include tracking public vehicles (buses, trams, subways and bikes for hire) or mapping bumps on the road to quickly notify authorities where to intervene [23].

The obvious difference between these sensors and the fixed position sensors is that they are always in motion with their owner. Formally, () the data from the GPS sensors can be expressed as an ordered series of measurements where each sample of GPS coordinates is connected to the time data:

$$\bar{p} = \{p_1 \rightarrow p_2 \rightarrow \cdots \rightarrow p_t\} (8)$$

$$t = 1,2, \ldots T (9)$$

p_t: latitude and longitude coordinates and a t timestamp.

In another sense, the data is expressed as a connected series of points linked to their time stamps, each of which contains latitude and longitude. Due to this property, data from moving sensors is also called space-time data where the space part is the GPS coordinate and the time part is the time stamp.

3. Traffic Applications

With the evolution of urban traffic, appeared new applications that contribute to the effectiveness of traffic networks:

3.1.Vehicle –to-everything (V2X)

V2X,Vehicle-to-Everything or Vehicle Connected to Everything is a communication technology that allows vehicles to connect to the mobile and fixed part of the road system that surrounds them. There are several components that make this technology usable and powerful, and promise more development in the future.

- **Vehicle-to-vehicle (V2V)**: The first element of this technology is the communication between two vehicles, which allows them to share crucial information that can allow them to prevent accidents and congestion.
- **Vehicle-to-infrastructure (V2I)**: Infrastructure is connected external systems such as street lights, buildings and even cyclists or pedestrians, and this component allows the vehicle to communicate with these systems. The future of this technology is promising due to improvements in everyday life and its rapidly expanding sophisticated capabilities.(V2X’s principal purpose is increasing safety and preventing collisions and this technology is applied in two types of vehicles)

In a traditional vehicle, the V2X is simply applied to transmit important environmental information to the driver, including weather conditions, nearby accidents, road conditions and dangerous activities of nearby vehicles.

In autonomous vehicles, the V2X provides additional information to the vehicle's existing navigation system. Security, automatic toll payment, parking and the like are all the benefits and goals of V2X technology. This technology uses short-range, weather and interference-resistant wireless signals to communicate with vehicles and infrastructure.

This technology uses short-range, weather and interference-resistant wireless signals to communicate with vehicles and infrastructure.

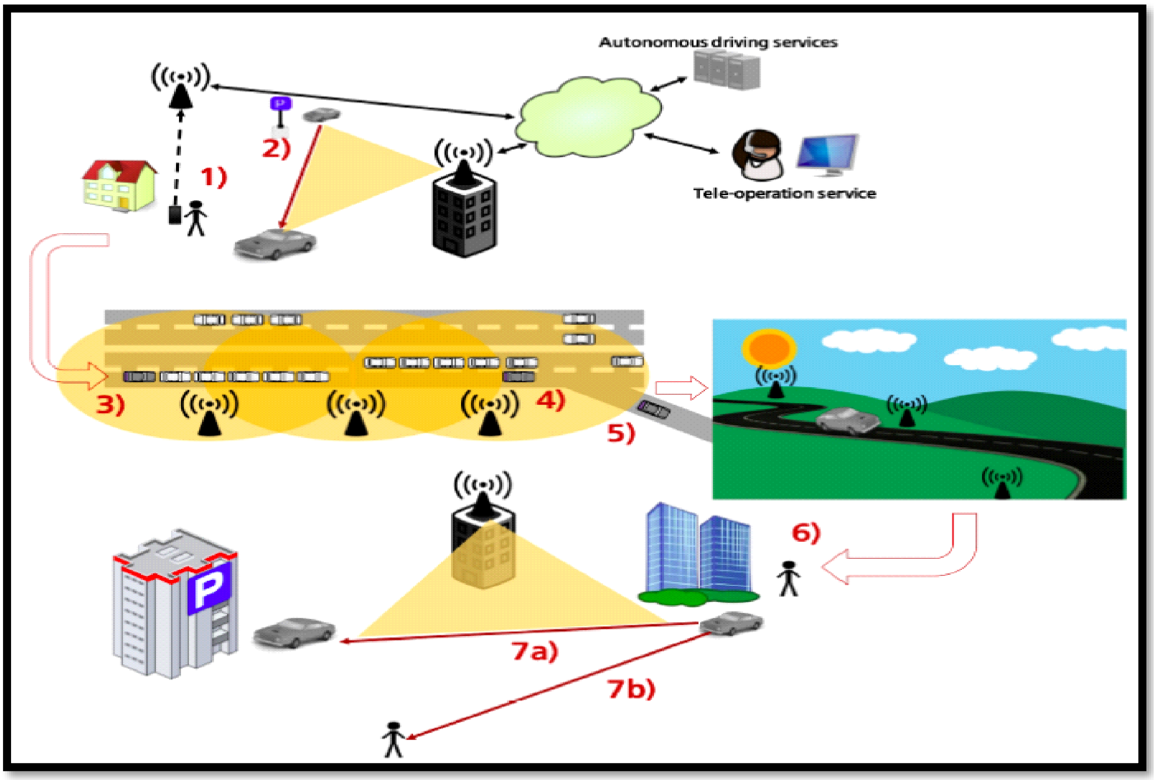


Figure 2: showing V2X technology with its connected components

- **Standards of V2X**

a. IEEE 802.11p

The initial V2X has been standardized on the basis of a Wi-Fi branch, IEEE 802.11p (which is part of the IEEE WAVE or Wireless Access for Vehicular Environments program), operating in the unauthorized frequency band 5.9 GHz. IEEE 802.11p, which was finalized in 2012, underpins dedicated short-range communications (DSRC) in the United States, and ITS-G5 in the European initiative Cooperative Intelligent Transport Systems (C-ITS).

V2X communication via 802.11p goes beyond limited visibility sensors such as cameras, radars and LIDARs, and covers V2V and V2I use cases such as collision warnings, speed limitation alerts limit and electronic parking and toll payments.

The functional characteristics of 802.11p include a short range (less than 1 km), low latency (2 m) and high reliability, according to the US Department of Transport, it "works in mobility conditions with high vehicle speed and offers performance insensitive to extreme weather conditions (e.g. rain, fog, snow, etc.). “Essentially, 802.11p expands a vehicle's ability to” see the environment around it, even in bad weather time.

b. Cellular V2X

The main advantage of the C-V2X is that it has two game modes, of which, cover most of the possibilities. The first is direct low-latency C-V2X communication via the PC5 interface in the unauthorized range of 5.9 GHz, which is designed for active safety messages such as instant road alarms and other v2V positions, V2I and V2P at short range. This is closely aligned with what IEEE 802.11p technology offers, which also uses the 5.9 GHz range.

Communications via the **Uu** interface on the regularly licensed band cellular network are **secon** mode, and it can handle V2N use cases such as infotainment and latency tolerant security messages for a range of longer road risks or traffic conditions. Due to non-use of cellular connectivity, IEEE 802.11p cannot correspond to this mode by establishing ad hoc connections with roadside base stations.

3.2.Vehicular Ad-hoc Network (VANET)

The ad-hoc vehicular network (VANET) is not a new subject like many technologies in our research, but there are still many research challenges and difficulties that have not been resolved so far, and researchers continue to develop it.

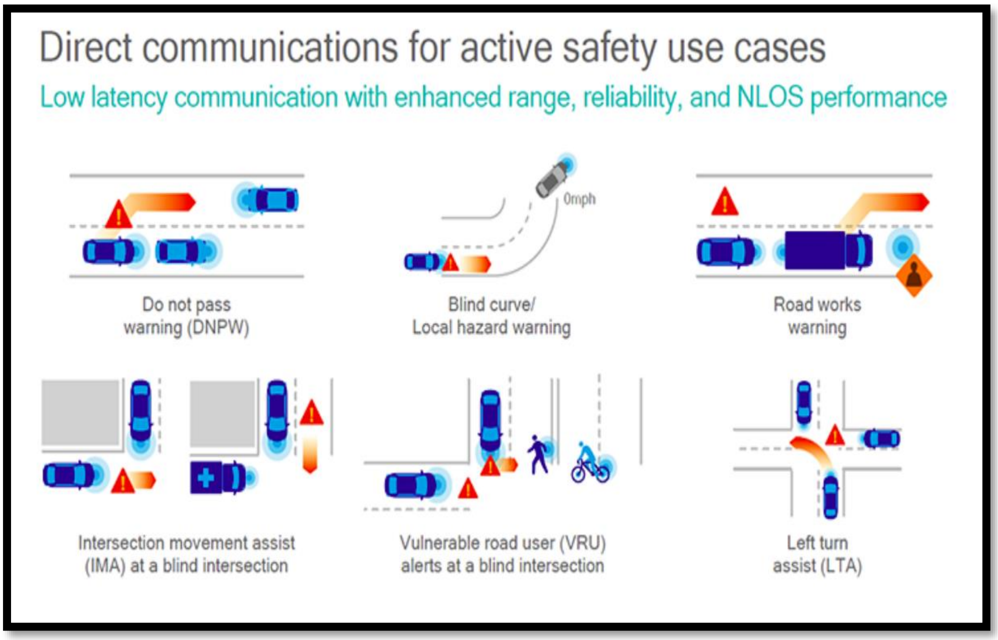


Figure 3: Direct communication for active safety use cases

As its name suggests, VANET is an Ad Hoc network, its main objective is therefore to create a vehicle network without the need for a central base station or any controller, and to maintain communication. The emergency Critical medical is one of the example applications of VANETS, where there is no infrastructure when it is essential to transmit information to save human lives.

Despite its applicability, there are new challenges and problems in complementing VANETs, which need to be resolved. control tasks, as well as their own communication needs..

3.2.1. History of Vanet

VANET has crossed many forms before reaching the version that is applied today, and we must discuss these forms before going into the details of VANETS. As shown in Figure 4, WANET is the parent field of all ad hoc networks. VANET is a brother of MANET which organizes its own communication system without any dependence on any other infrastructure. The most common use of MANET is in the military due to its simple and basic communication method, as well as the sharing of data between different computers. VANET is similar to MANET with some modifications. VANET includes mobile nodes (MN), road units (RSU). Mobile nodes are the sensors built into vehicles which are called on-board units (OBUs) for signal processing (data sharing) to and from the RSUs. RSUs are installed fixed units which are the gateway for communication between MN and the servers or the Internet. There are many services provided by VANET, but the most important of all is road safety services for the reduction of road accidents by sharing data via the Internet.

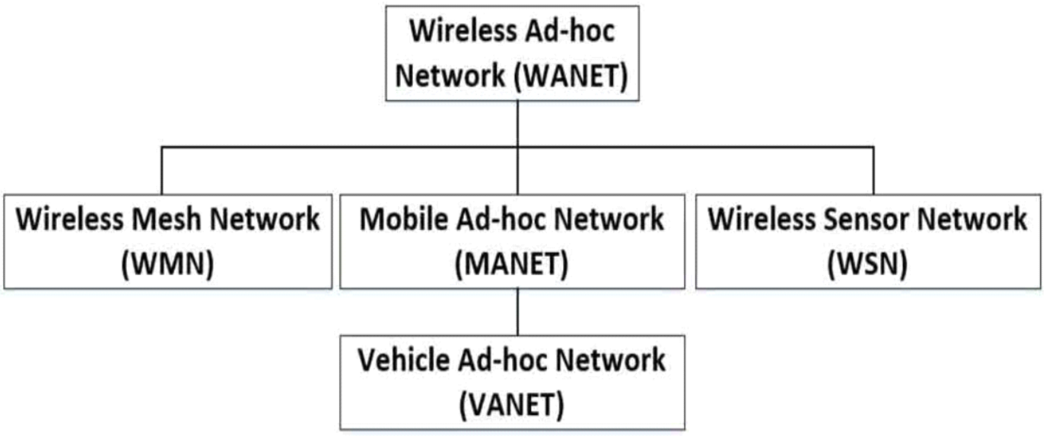


Figure 4: Classification of mobile Ad-hoc Networks

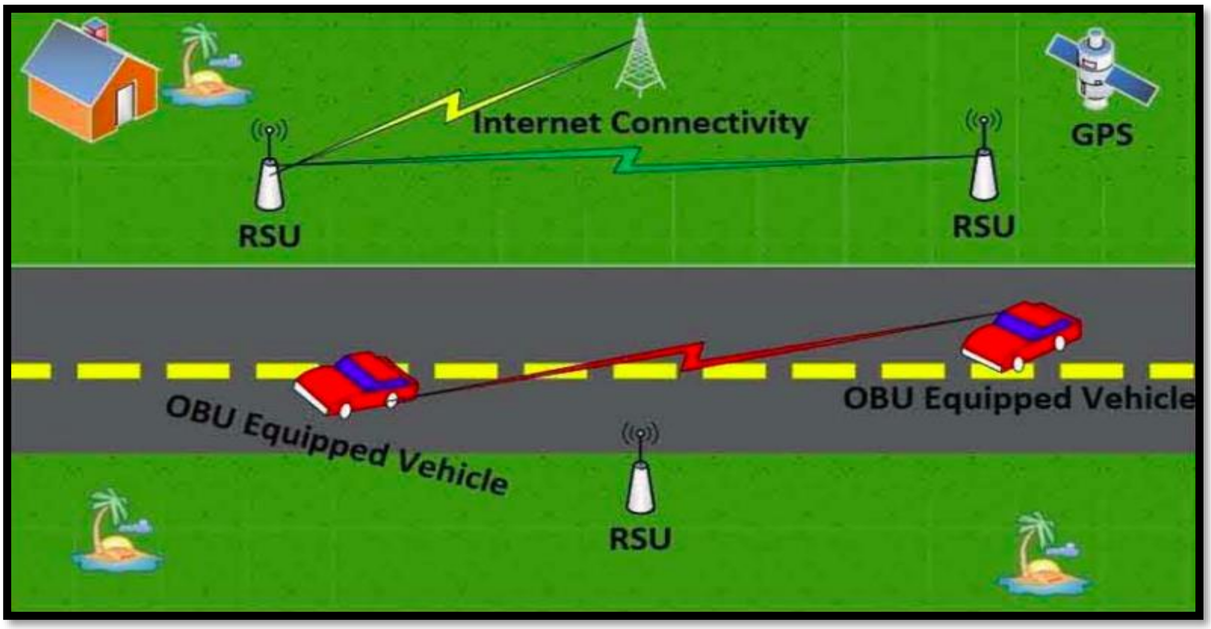


Figure 5 :VANET Communication System[111]

3.2.2. Architecture and Network Modeling

Basically, VANET does not have a specific architecture or topology that must be followed. Despite this, VANET generally involves moving vehicles which must communicate with each other and with nearby RSUs. The main difference between VANET and MANET is the paths followed by the nodes, in MANET, these paths are random, but in VANET the vehicles follow fixed roads and highways. (VANET is part of MANET, but it is also an individual research field, particularly in terms of network architecture design). In the VANET architecture, an on-board unit (OBU) in a vehicle consists of a wireless transmitter and receiver.

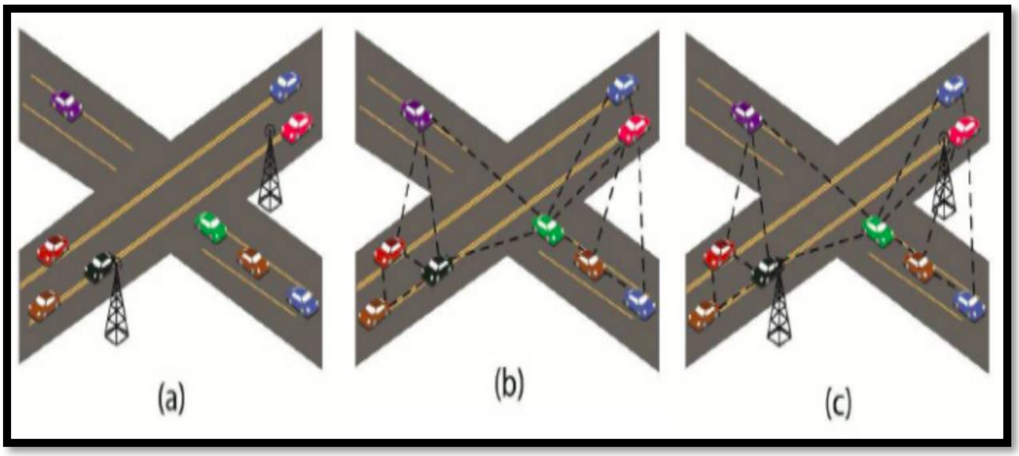


Figure 6 :some network architectures in VANET[111]

There are three major communication architectures that are commonly used in VANETs:

- **First architecture**: All vehicles communications pass through RSUs, this architecture can be seen as a wireless local area network (WLAN).
- **Second architecture**: There is no need for RSU in this architecture, because all the communications are directly established between vehicle. This can be classified as Ad-hoc architecture.
- **Third architecture**: Is a hybrid architecture between the first two models, where some vehicles communicate directly with each other and while others may need an RSU to communicate. [26]. FIG6 shows these three possibilities.

Understanding network architecture is important in order to realize the full potential of vehicular communication. According to most of published studies [27, 28], VANET scenarios are divided into three categories: Urban, Rural and Freeway/Highway.

Ensuring that network interconnection needs for an entire vehicle environment are covered is one of the main reasons to investigate this way. The challenges differ from one environment to another and all must be overcome. For instance, in a sparse network like highways, low vehicle density remains the main problem. Even in some urban environments, low traffic volume at hours of darkness can cause long delays on the network.

The high mobility of nodes in ad hoc wireless networks is a crucial attribute in this type of architecture, and this attribute makes modeling the communication scenario difficult and complicated. In order to model a VANET environment, we need to deepen the key characteristics of vehicle mobility such as acceleration, deceleration, lane changes and human driving habits. Much research [29,30,31] has been done to include mobility in the design of VANET in order to explore these characteristics. In the vehicular networks, the traffic mobility model must includethe behavior of moving vehicles individually and in groups for efficient and error-free packet transmission

3.3.Intelligent Traffic Management

With the increasing number of private motor vehicles, cities are becoming more congested and their surveillance becomes more difficult and complicated. This problem exhausts infrastructure managers around the world. Congestion and traffic accidents cause significant loss of time, property damage and environmental pollution, making it one of the biggest concerns. There is an urgent need to improve traffic management due to the financial losses caused by traffic congestion. The appearance of the Internet of Things (IoT) offers a new trend in the intelligent development of traffic.

Using IoT, agents, and other technologies can improve traffic conditions and reduce traffic pressure. The information generated by IoT traffic and collected on all routes can be presented to travelers and other users. The system can recognize current traffic, traffic conditions and future traffic thanks to traffic data collected in real time. The system can generate current and real-time traffic information to help drivers choose the best routes. The system can therefore precisely manage, monitor and control moving vehicles. Building an intelligent traffic system based on IoT has a number of advantages such as improved traffic conditions, reduced congestion and management costs, where there is no room for new roads, and many of the simplest ways to widen roads have already been used. Although all of the grassy central medians designed along freeways have been paved inward, many urban highway corridors in the United States still have ways to extend outward and replace the slopes with retaining walls. A recent study into the feasibility of expanding major highways in the Los Angeles area found that about 118 miles by 136 miles had space in the existing reserve or required little land purchase to develop. Reliability, road safety and independence from bad weather.[32] [33]

The traffic IoT must include all traffic elements such as roads, bridges, tunnels, traffic lights, vehicles and even drivers. All of these elements are connected to the Internet for practical identification and management by recognition devices such as RFID devices, infrared sensors, global positioning systems, laser scanners, etc.

Traffic IoT provides the acquisition and integration of traffic information and supports the processing and analysis of all categories of traffic information on the streets of a large area automatically and intelligently. This is how modern traffic management turns into an intelligent traffic system based on IoT.

4. Traffic Dataset Types

Traffic data differs with the sources of this data, for that we distinguish thre types of data.

4.1.Scalar-Based Datasets

The simplest data model to predict traffic flow uses data from fixed position sensors without pretreatment. As mentioned in equation (6), each $m_{p,\,t}$ measure contains a scalar value and the time stamp of the measurement at position P. Data from mobile sensors can also be used in this model, but measured GPS lane data must be preprocessed, which means values only for those examined P positions are extracted. The purpose of the traffic prediction is to determine the value $m_{p,}\ t\text{-}1$ based on m_p, where p determines the position. Ordered sequences of scalar values can be modeled according to time series, which are a known data model.

The time series can be characterized by the following characteristics. The trend is a gradual increase or decrease in values in the series over time, which may be global, local, or both. It can also be linear or non-linear. A seasonal cycle is a repetitive and predictable pattern in series values, while a seasonal non-cycle repeats, which can be unpredictable. Variations in time series values that cannot be explained by the model are called residuals.

The main drawback of using a scalar model is that it masks the spatial correlations and only describes the temporal correlations. Sometimes we only care about the flow at a certain point along the way. The scalar model is the right approach for this. In most cases, however, we also want to understand the behavior of the flow along the entire path, which is not possible with this model.

Scalar models also have scalding problems because calculating forecasts for each sensor in a large sensor network is an intensive computing task.

4.2.Vector-Based Datasets

Instead of using a scalar value to characterize the flow at a point in space and time, vector models define a vector that describes the state of circulation (also called state vector). As input data sources for vector models, fixed position sensors (if it can be assumed that enough sensors are installed on roads) or moving position sensors or a mixture of them. With regard to the spatial factor, it is distinguished between the univariate and multivariate versions.

A univariate model observes a sensor, while the multivariate versions observe more sensors.

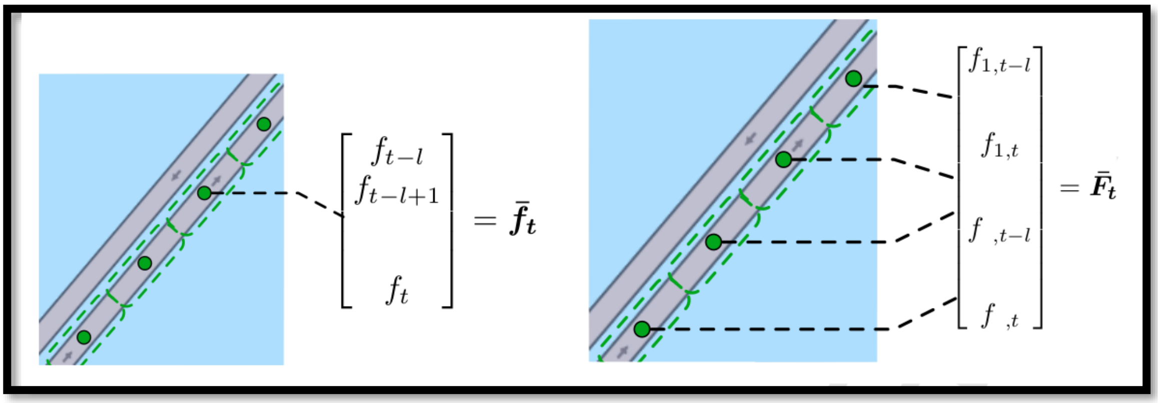


Figure 7: Univariate and Multivariate models[109]

- **Univariate Vector Models:**

The figure 7 show a univariate vector model for a given sensor of the transportation network, where the state vector of the current flow at time *t* can be defined as:

$$\bar{f} = \{f_t - l\,; f_t - l + 1 \ldots, f_t\} \quad (10)$$

Where l (lag) is the number of time intervals handled by the model. f $_{t-1}$ denotes the measured value at time t − l. The goal is to predict the next f $_{t+1}$ value.

In most cases, authors use separate time intervals [58], but some works use overlapping intervals. To give an example of overlapping time intervals, we leave 2 and assume that we use time intervals of 5 minutes. Then f t indicates the actual interval, f t-1 indicates the previous 5-minute interval, and f t-2 indicates the previous 10-minute interval. Unfortunately, the univariate vector model, like scalar models, is also unable to describe spatial correlations.

- **Multivariate Vector Models :**

In a multivariate vector model (Figure 1b), we observe more sensors of a transportation network. Thus, the state vector of the current flow at time t can be defined as:

$$\bar{f}_{t=}\{f_1\,, t - l, \ldots, f_{1,t}, f_{2,t} = l \ldots f_{j,t}\} \quad (11)$$

Or

$$\bar{f}_{t=}\{f_1\,, t - l, \ldots, f_{j,t} - l, f_{1,t} - l + 1, \ldots, f_{j,t}\}(12)$$

Where l (lag) is the number of time intervals processed by the model, and j is the number of sensors present in the state vector. The purpose of traffic prediction here is to determine the value of f x, t + 1, where x is an arbitrary sensor and t + 1 is the next time interval. As you can see, the order of values for multivariate vector models can vary.

4.3.Matrix-Based Datasets

In a matrix model, matrices are defined that describe the current state of the flow. Fixed position (if it can be assumed that enough sensors are installed on the streets), mobile position sensors or a mixture of them can be used as data sources. Unlike scalar or vector models, matrix models always identify spatial and temporal correlations. These models can be classified as a macroscopic model, which means that instead of examining a point in space in detail, they are able to identify correlations between larger areas.

In addition to the KNN prediction model [34] (or its extended variants), the most popular prediction models used for matrix models in related work are the convolutional neural network (CNN) [35], defined neural networks. by the user [36] and Bayesian networks [37]. Based on the properties of matrix models, they can be divided into two main groups: spatio-temporal matrix models and regional matrix models.

- Time-space Matrix Models
- Region Matrix Models

4.4.Environmental and Seasonal Information

Traffic flows are not only influenced by the cardinality and the behaviors of their entities, they are also significantly affected by other factors such as weather, seasonality (holidays, day of the week, seasons, school hours), events, road construction, air quality or lighting conditions. Take for example heavy rain, which could lead to overcrowded sections, because pedestrians avoid open spaces and move under underground passages and covered areas. Another case is one where there is a high risk of accidents, such as poor vision or slippery roads which make car drivers slower than usual, which increases the likelihood and volume of traffic jams and congestion [24].

Unusual conditions produce remarkable changes in traffic flows, these conditions can be observed on different days of the week as well as in seasonality, in addition to this, holidays can also play a major role and cause heavy loads on the road network and cause severe congestion. Take the example of roads and walking paths near universities, which are considerably more burdened during the school period than during the exam period or the summer break, and this proves the effects of the seasons of the year on debits.

All of these examples demonstrate the critical importance of these external factors such as weather forecasts and seasonal impact, and the need to use their data sources. By ignoring them, our forecasts would not be precise and refined [25], and huge inaccuracies could appear jeopardizing the applicability of forecasting traffic flows.

5. Traffic Data Modeling

Traffic modeling aims to accurately recreate the traffic observed and measured on the street. Traffic modeling assumed the appearance of a traffic system without eplication. It was developed on the basis of the experience of the modeller who integrates mathematical models in the traffic system [38]. Traffic modeling plays an important role in traffic engineering. It can be applied to plan and manage traffic within certain road networks [39]. For example, creating smooth traffic at an intersection, etc. [40]

Primarily, simulation models specialize in three output values to solve traffic problems [41]. The first is that the traffic is fluid. In traffic flows, alternative routes are often identified based on the quantity of vehicles. Using the simulation model, the modeller can imagine a way to reduce the degree of congestion on certain roads. The second output is that of the network element. The network element in traffic simulation includes link, merge, cross link and other route elements [42]. this can be associated with the geometric layout of the road. With the help of appropriate simulation software, the geometric design of the roads is often modified to determine how it can influence the current traffic situation. The third exit is that of the skim category. The simulation model can help estimate the time and price of the trip. this can be particularly used when the evaluation of traffic improvement needs to be measured. Transport planner can easily compare performance without additional cost in time and money.

6. Traffic Performance Indicator

Cities today face many common traffic problems and implement similar urban traffic mnagement solutions, with intelligent transport systems (INTELLIGENT Transportation, ITS) playing a leading role. However, it is extremely difficult for a city to objectively assess the impact of specific policies and technologies and to use lessons learned from other cities without a set of widely accepted criteria and methods.

We discuss the peak hours factor, the reliability of travel times, the number of services and a measure of road efficiency.

6.1.Peak Hour Factor

The peak hour factor (PHF) is a possible indicator of fluctuations in traffic flow during peak hours and periods of high current. It is calculated for a day as the average flow during the hour with the maximum flow, divided by the peak current for a quarter of an hour during this hour [44]:

$$(13)PHF = \frac{\bar{q}|60}{\bar{q}|15}$$

Suppose we measure the currents on a main three-lane road during a morning rush hour: from 7:00 a.m. to 8:00 a.m., we measure 3,500, 6,600, 6,200 and 4,500 vehicles per hour in a quarter. The average total flow q / 60 is 5,200 vehicles per hour, with a maximum flow of 15 minutes q / 15 - 6,600 vehicles per hour. The PHF thus corresponds to 5200/6600 - 0.78.

6.2.Travel Times And their reliability

When traveling, people like to know how long a specific trip will take (for example, by public transport, car, bicycle, etc.). This notion of expected travel time is one of the most tangible aspects of the trip as it is perceived by travelers. When people get to work, they have to arrive at their destination on time. Based on this premise, we can naturally say that people reason with an integrated safety margin: they consider the average time it takes to reach a destination, and use it to decide their departure time.

In addition to the obvious human logic mentioned above, there is also increased interest in accurate travel time information in the context of Advanced Traveler Information Systems (ATIS). An essential ingredient here is the accurate forecast of future travel times. In addition to accident detection, drivers can, for example, receive correct journey time information so that they can stay informed of actual traffic conditions and possibly change their route. The requested information can be reached by a mobile phone (for example depending on the mobile phone provider), it can be broadcast by radio (for example the Traffic Message Channel - TMC), or accessible via certain road sections (for example, dynamic route information panels (DRIP), etc.

A driver's journey time to the end of a journey can be defined as the time required for an intermediate route between two sites. "In this context, the dynamic travel time experienced, from a specific time t0, is plotted on a street section of length K defined as follows [46]:

$$T(t_0) = \int_0^k \frac{1}{v(t,x)} dx \forall t \geq t_0 \quad (14)$$

For which it is assumed that all the instantaneous local speeds of the vehicle v (t, x) are known at any point on the route and at all times (hence the term dynamic travel time).

In most cases, however, we do not know all of the v (t, x), but only a finite subset of them, which is defined by the location of the detection stations. Travel time can then be approximated using the speeds recorded at the start and end of a section (here, there is an underlying assumption, that vehicles move between the detector positions at a speed more or less constant). As already mentioned, the driving time experienced requires knowledge of the local vehicle speeds at all times after T0. As this is not always possible, a simplification can be used, which leads to the so-called experienced journey time:

$$\tilde{T}(t_0) = \int_0^k \frac{1}{v(t,x)} dx \quad (15)$$

6.3.Level of service

Historically, one of the most important performance indicators for assessing the quality of transport operations was the level of service (LOS) introduced in the 1960s. It is represented as a classification system with one of the six letters (AF), after which LOS A designates the best operating conditions and LOS F the worst. These LOS measurements are based on road characteristics such as speed, journey time and the perception of comfort, comfort, [47]. As is common among traffic engineers, these statistics, representative of these characteristics, are collectively called efficiency measures (MEO).

Levels A to D are representative of the free flow conditions, in which LOS A corresponds to free flow, LOS B for reasonable freedom of flow, LOS C for stable transport operations and

LOS D for transport operations adjacent unstable transport. LOSE recalls almost unstable flow conditions near capacities, while LOS F corresponds to overloaded flow conditions (caused by structural or accidental overload) [44].

As an example, we give an overview of the different service levels in Table I (based on [44].

LOS	Density (veh/km)	Occpancy (%)	Speed (km/h)
A	0 → 7	0 → 5	≥ 97
B	7 → 12	5 → 8	≥ 92
C	12 → 19	8 → 12	≥ 87
D	19 → 26	12 → 17	≥ 74
E	26 → 42	17 → 28	≥ 48
F	42 → 62 > 62	28 → 42 > 42	< 48

Table 1 : LOS indicators levels

The calculation of service levels can be done using a multitude of methods; some examples include the use of density (on highways), the use of average space speed (in arterial streets), the use of delay (at signposted and unsignalized intersections), etc. [47].

6.4.Efficiency

In the work of Chen et al [48], stresses that the main reason for congestion is not that demand exceeds capacity (i.e. the number of passengers wishing to use a certain part of the transport system exceeds the capacity of available infrastructure), but that it actually does this is inefficient highway operation in times of high demand. To quantify this efficiency, they first examine the speed that prevails when a highway operates at its maximum efficiency, i.e. the highest flow rate (corresponding to the actual capacity, which differs from the capacity of the HCM, which is calculated from the physical properties of the road). Based on the distribution of 5-minute data samples from approximately 3,300 detectors, they examine the velocity in periods of very high throughput. This leads to a so-called continuous speed compared to only 60 miles per hour (which corresponds to 60 mph - 1,609 - 97 km/h).

The performance indicator they offer is called efficiency η and it is based on the ratio of the total number of kilometers driven by the vehicle (VMT), divided by the total number of hours driven by vehicle (VHT). Note that since the units of VMT and Vsust must match, we suggest using the terminology of total distance traveled by the vehicle (VDT) instead of VMT, in order to eliminate any possible confusion. VDT and VHT are defined as follows:

$$VDT = qK \qquad (16)$$

$$VHT = \frac{VDT}{\bar{v}_s} \qquad (17)$$

With, as before, q-Flow, K the length of the road section and compared to the average space velocity. With the definitions above, we can see the effectiveness of a section of road like:

$$\eta = \frac{VDT/VHT}{\bar{v}_{sust}} \qquad (18)$$

7. Traffic Flow Theories

Traffic is a complex system with similarities to the flow of physical particles. Successful applications of traffic physics include, for example, a hydrodynamic model of traffic.

A single vehicle can be treated as a particle flowing in a pipe. The compatibility of the physical approach with traffic theory has attracted many physicists for decades. Since many other collective physical systems are in an emerging state, traffic jams occur. Three-phase traffic theory developed by Russian physicist Boris Kerner explains congestion by phase transition in the traffic system

7.1.The meaning of phases

The concept of "phases" was originally used in fields such as thermodynamics, physics and chemistry. In these systems, "phases" denote different states of matter (for example solid, liquid or gaseous; or different compositions of materials in metallurgy; or different collective states in solid state physics). If certain "control parameters" such as the pressure or the temperature in the system are modified, the global state can also change if the transition is abrupt, we speak of first order transition, otherwise we speak of second phase transitions order if the transition is continuous.

When transferring these concepts to traffic flows, the researchers differentiated between single-phase, two-phase and three-phase models. The number of phases is mainly related to the stability characteristics (in) of traffic flows (i.e. the number of states that distinguish the instability diagram).

7.2.Three-phase traffic theory

In three-phase traffic theory, the three traffic phases are made up of free flows and two congestion phases: synchronized flow and large mobile congestion. The three phases offer qualitative characteristics of traffic congestion. Due to the complexity of the transport system, such as. With regard to driver inhomogeneity and behavior, however, the theory is limited to quantitative correspondence with reality, although the most recent simulations may have shown qualitative characteristics.

In traffic theory, there are a large number of parameters (safety distance between vehicles, average length of vehicles, acceleration time). Among these, it suffices to provide only three parameters to understand the empirical properties of three phases: speed v, density and speed q of the vehicles. They are not independent from each other. You are bound by a simple intuitive expression.

$$q = \rho v \quad (19)$$

In other words, as long as both values are given, the other can be ignored. However, the correlation of the three complete variables is important to distinguish the three phases of traffic.

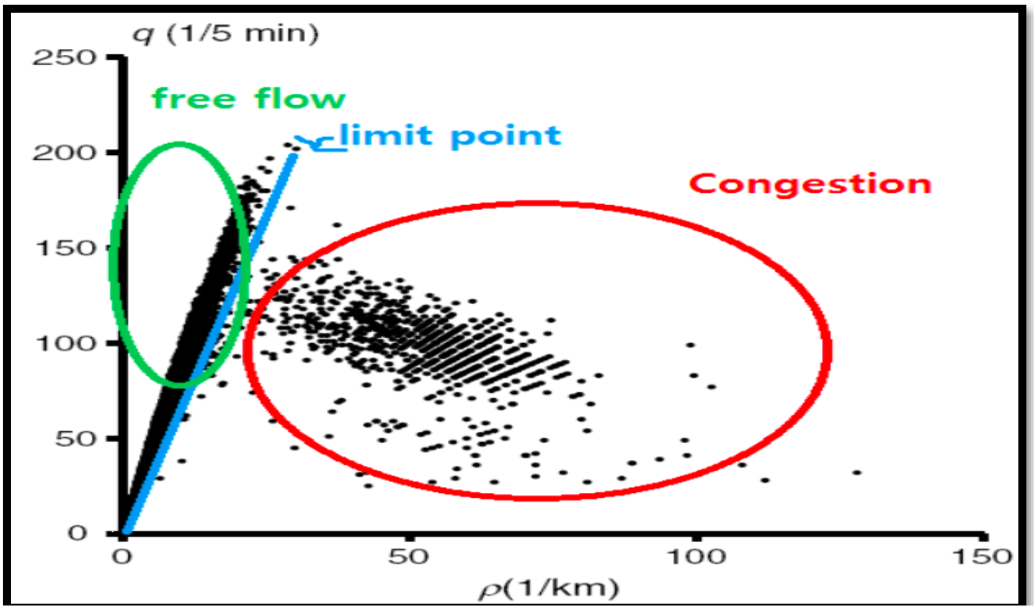


Figure 8 : Diagram showing the three phases (data taken from Japan highway). [49]

7.2.1. Free flow and Congestion

Free movement and traffic jams are more intuitively different than the three phases. Free flow and congestion can be easily defined using the basic diagram in the density flow level. In Figure 8, the points are divided into two areas. a positive slope corresponds to free flow. In free flow, since there is no significant drop in velocity, the flow is almost proportional to the density (its tendency tends to decrease with increasing density). This maximum point in the basic diagram is called the limit. Another set of points corresponds to traffic jams. Traffic jams mainly occur at the bottleneck. In a traffic jam, the average speed of vehicles decreases. In addition, the variance of points in traffic jams is much greater than the free state. [50]

In free circulation, vehicles are not (much) influenced by each other and can move freely. In multi-lane traffic, this means that vehicles can pass freely.

7.2.2. Synchronized Flow

The vehicle speed decreases significantly in the synchronized flow, but no notable change in the flow is observed [50]. this is due to the increased density of the vehicle so that the product of speed and density remains almost the same. the term synchronized reflects the synchronization of vehicle speed in different lanes. The downstream front is mainly connected to the bottleneck. Synchronized flow can be divided into three models, depending on the development of the downstream fronts and the upstream front. [51]

We distinguish three types of synchronized flow:

- **localized synchronized flow**: the downstream front is connected to the bottleneck. The front oscillation, but the average width of the pattern does not change.
- **widening synchronized flow**: the downstream front is attached to the bottleneck, the upstream front spreads continuously to the rear.
- **moving synchronized flow**: a whole pattern is propagated, but cannot penetrate the adjacent bottleneck (capture effect). Capture effect distinguishes mobile synchronized stream with wide mobile congestion [52].

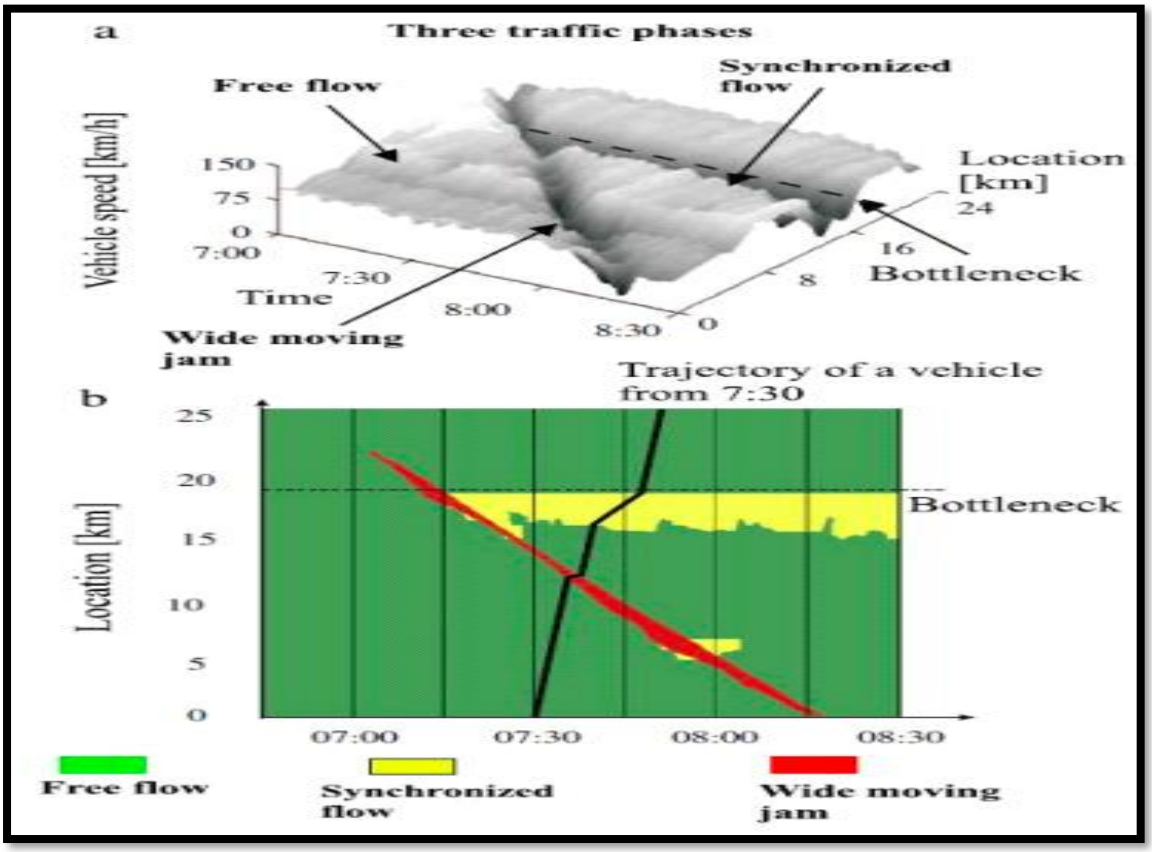


Figure 9 :A diagram showing the three phases in the space-time plan.49]

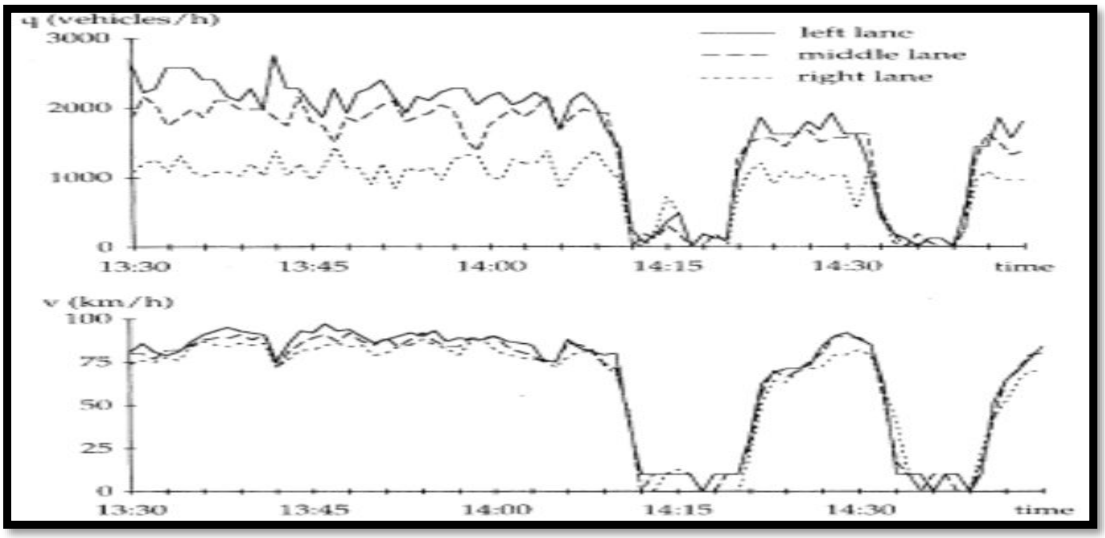


Figure 10 :a diagram showing the rapid drop of wide moving jam[53]

7.2.3. Wide Moving Jam

A large mobile packing can only take place spontaneously thanks to a synchronized flow. At this point, the flow and speed decrease considerably and become relatively uniform with the synchronized flow. Therefore, there is empirically a large variance in the density flow level from a synchronized flow. Spreads backwards with average speed vg. this can happen through the next bottleneck. The line formed by points in the density flow plane is called line J. The slope of line J corresponds to the speed at which the pattern vg propagates.

Conclusion

In this chapter dived into vehicular traffic notions, and we tried to understand the basics of urban traffic flow, and its challenges, in order to do that we had to understand the nature of its data, how to collect it and the methods used to model it, before that we talked about the sources of traffic data and the various technologies used.

In the next chapter we will précis our problematic, and discover the techniques used to attain our objective by comparing used techniques.

Chapter 02

Deep Learning Architectures For Traffic Flow Prediction: State Of Art

INTRODUCTION

In the literature, there are many prediction models that are used to predict traffic flow. However, the prediction must be defined as it is a fundamental property of traffic flow that affects the choice of the prediction model.

According to [54], the forecasting of the traffic flow indicates the possibility that the prediction satisfies certain accuracy requirements over a desired forecast time horizon. In other words, what is the maximum time that can be predicted by a model with a limited error?

The actual value of a prediction consists of a predictable component and an error [55], which includes both the prediction error and the unpredictability of the uncertainty. The predictable value is derived from the deterministic part, and the predictable part of the uncertainty depends on the ability of the model to predict the uncertain part of the traffic flow with the required precision. Uncertainty is influenced by many factors, such as weather, days of the week, events, road construction, lighting conditions, etc. Integrating external environmental factors and merged data [56] into the model is crucial to reduce forecast errors and increase the predictable part of the uncertainty. The predictability of traffic flows is directly linked to the time horizon of the forecast. Intuitively, forecast accuracy decreases with increasing forecast horizon [57].

In this chapter we will discuss the methods and techniques used to predict the state of traffic, we will begin by mentioning the common used methods in this field and distinguishing them by their characteristics and types, after that we clarify the method that interests us which is deep learning, and in the end of this chapter we will talk about two chosen deep learning algorithms and understand the basic notions of them.

1. Deep Learning For Traffic Forecasting

Neural networks are the main model for predicting traffic flow these days; their main property is their ability to model non-linear, stationary / non-stationary behavior, in addition to their extensibility. This means that the spatio-temporal property can also be taken into account and that environmental data sources can be easily integrated. All of these properties reduce the unpredictable part of the uncertainty, which could also significantly reduce the prediction error in extreme cases.

NNs have long been used to predict time series in which data is modeled using scalar models. The authors began to study traffic flow as time series using the Feed Forward Neural standard. Networks (FFNN) with a backpropagation algorithm [102].

Backpropagation is a monitored learning algorithm using Descent artificial neural networks. In the case of an artificial neural network and an error function, the process calculates the error function with respect to the weights of the neural network. These NNs can work better than simple parametric models. However, they cannot take advantage of the spatio-temporal property of traffic flows.

Here are some New York algorithms almost used in traffic prediction

- **Time Delayed Neural Networks (TDNNs)**: augment the input scalar model with delayed copies
- **Recurrent Neural Networks (RNNs)**: retain an internal state (memory) by using a directed cycle in neurons, but they have problem of long-term dependancy [103]
- **Long-Short Term Recurrent Neural Networks (LSTMs)**: are the predecessors of RNNs because of their capabilities of long and short term dependency learning[104]
- **Gated Recurrent Units (GRUs)**: have the same capabilities as RNNs with a simpler architecture with fewer parameters
- **Convolutional Neural Networks (CNNs)**: with the using of convolution layers, they are capable of extracting spatial correlation over a map [105,106,107]

2. TrafficPrediction Techniques

Forecasting traffic is a vast field with many variables, we will mention them by their categories and types

2.1.ParametricModels

2.1.1. Traffic Simulation Models

Traffic simulation models are mathematical models that help plan and design transportation systems. Instead of using historical and real-time traffic data, these models simulate traffic; because in the design phase of a road network, no historical data is available. It is important to emphasize that future traffic levels can be estimated using traffic simulation models to validate the relevance of the design of a transport system; however, these models are not capable of predicting the next traffic state based on historical data and in real time.

The basic elements of traffic simulation models have been established by Beckmann, McGuire and Winsten [58]. In these models, traffic is simulated by an origin-destination (OD) matrix, which describes the movement of vehicles in a certain area. The OD matrix includes a cell representing the number of journeys from the origin (line) to the destination (column).

Traffic simulation models can be classified according to their scope as microscopic, macroscopic or mesoscopic.

In the microscopic view, each individual actor in the street and their interactions are modeled in a multi-agent system [59], in which each agent keeps a record of his trip, including basic information or behaviors (for example, lane change behavior or acceptance of distance behavior, if conflicts with other vehicles may arise). A typical microscopic simulation approach is that of cellular automata (CA), in which the streets are divided into cells that can be empty or occupied by a vehicle, and time is dissected in the reflection phases of summer. CA has the ability to reproduce a wide range of different traffic phenomena and, due to the simplicity of the model, it is numerically very efficient. This model was combined with OD forecasts to obtain network-wide traffic forecasts [60].

In the macroscopic view, only the total variables of a road network are taken into account, e.g. B. density, speed or number of traffics if these variables are determined for each road segment of the road network [61]. There are two methods of allocating data traffic to the simulated road network: static and dynamic allocation.

Mesoscopic models are combinations of macroscopic and microscopic models [62]. First, traffic is assigned to different road segments using macroscopic models, after which individual cars are moved across the network based on the calculated microscopic traffic variables. The great advantage of the mesoscopic approach is that a greater variety of phenomena can be modeled, such as the operation of installed traffic lights, highway mergers, weaving sections or lanes reserved for busy vehicles.

2.1.2. Time SeriesModels

The basic time series models assume that the value of the series at time t depends linearly only on its previous values with added random noise [51]. The autoregressive (AR) and moving average (MA) components are used to model the time series, thus forming a moving average autoregressive model (ARMA). AR predicts the variable of interest using a linear combination of past values of the variable, while MA uses past forecast errors in a regression model. Since most time series have non-stationary behavior in practice, the ARMA model can be generalized to manage non-stationarity byapplying differentiation (calculating the differences between consecutive observations). This extension of the ARMA model is called an autoregressive integrated moving average (ARIMA). The disadvantages of using time series models are that they cannot handle non-linear processes, and it is difficult to integrate environmental data sources into them.

2.2.Non-ParametericModels

Non-parametric models assume that the distribution of data cannot be defined as a limited set of parameters, but they can often be defined by taking an infinite dimension. Therefore, more data is generally needed than for parametric models. Nonparametric models are more flexible than parametric models because the amount of information that data can collect can increase with the amount of data. The advantage of these models is that they are capable of managing nonlinear dynamic processes and can also use spatio-temporal relationships. Some of them are also capable of integrating environmental data sources, which can increase the accuracy of forecasts in extreme cases. The downside of nonparametric models is the training of the model or the prediction itself, which can be a computer intensive task, is the comparison with parametric models, since huge amounts of data have to be processed.

2.3.Machine Learning Techniques

2.3.1.Bayesian Network

Bayesianprobability

Is an interpretation of the concept of probability, which can be interpreted as a reasonable expectation representing a state of knowledge, rather than the frequency or prosperity of a phenomenon or as a quantification of a personal belief.

The Bayesian interpretation of probability can be seen as an extension of propositional logic which allows us to think with hypotheses, that is to say with propositions whose truth or lie is unknown. In the Bayesian view, a hypothesis is assigned a probability, while in the frequency inference rule, a hypothesis is generally tested without a probability being affected.

The Bayesian probability is classified as the probability of proof; To assess the probability of a hypothesis, the Bayesian probabilistic gives an earlier probability. This visit will then be updated in the light of new useful data for a later probability.

Bayesian inference

Is a method of statistical reasoning that uses the Bayesian theorem to update the likelihood of a hypothesis when more evidence or information becomes available.

Probability

The joint probability distribution of random variables $A_0, A_1, \ldots, A_n$, denoted as $P(A_0, A_1, \ldots, A_n)$, is equal to $P(A_1 \mid A_2, \ldots, A_n) * P(A_2 \mid A_3, \ldots, A_n) * \ldots *$

P(A_n) by the chain rule of probability. We can consider this a factorized representation of the distribution, since it is a product of N factors that are localized probabilities.

$$P(\cap_{k=1}^{n} A_k) = \prod_{k=1}^{n} P\left(A_k \middle| \cap_{j=1}^{k-1} A_j\right) \quad (20)$$

Conditional independence between two random variables, A and B, given another random variable, C, is equivalent to satisfying the following property: P(A,B|C) = P(A|C) * P(B|C). In other words, as long as the value of C is known and fixed, A and B are independent. Another way of stating this, which we will use later on, is that P(A|B,C) = P(A|C).

- **The Bayesian Network**

Bayesian networks are a kind of probabilistic graphical model that uses Bayesian inferences for probability calculations. Bayesian networks aim to model conditional dependence and therefore causality by displaying conditional dependence across the edges in a directional diagram. Thanks to these relationships, the return of random variables in the diagram can be done efficiently using factors. They can be used for a variety of tasks, including prediction, anomaly detection, diagnosis, automated comprehension, reasoning, time series prediction and uncertainty decision-making.

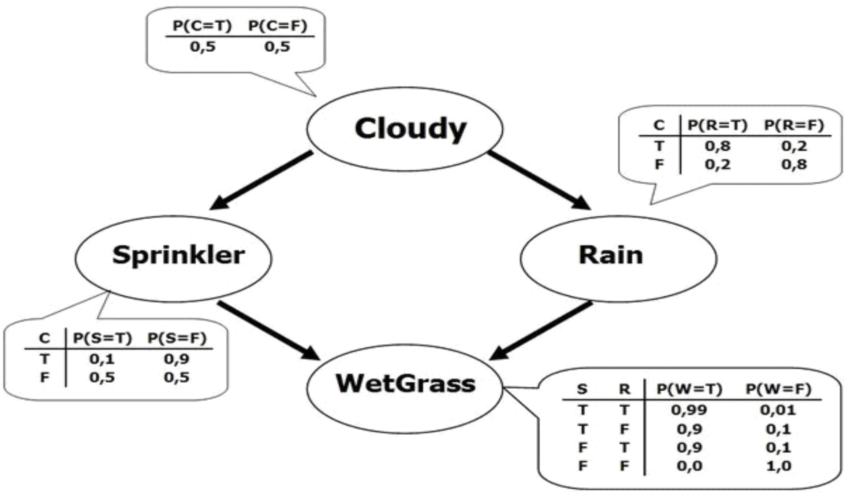


Figure 1 : Example of a Bayesian network

Connecting the random variables, A and B, this means that P (B | A) is a factor in the joint probability distribution, so we must know P (B | A) for all the values of B and A in order to proceed with the 'inference. In the example above, since Rain has an edge entering

WetGrass, this means that P (WetGrass | Rain) will be a factor, the probability values of which are specified next to the WetGrass node in a conditional probability table.

Bayesian networks fulfill the local Markov property, which states that a node is conditionally independent of its non-offspring with respect to its parents. In the example above, this means that P (cloud watering, rain) - P (cloud watering) because the sprinkler is conditionally independent of its non-falling rain, when it is cloudy. This property allows us to simplify the common distribution obtained with the chain rule in the previous section to a smaller form. After simplification, the common distribution for a Bayesian network is equal to the product of P (node / parent (node)) for all nodes, see below:

$$P\left(X_{1,\dots\dots}X_n\right) = \prod_{i=1}^{n} p\left(X_i|X_i, \dots, X_{i-1}\right) = \prod_{i=1}^{n} P\left(X_i|Parents(X_i)\right)(21)$$

In larger networks, this property allows us to significantly reduce the amount of computation required since most nodes generally have few relatives in relation to the overall size of the network. [66]

- **Notation**

Variables are represented with upper-case letters (A,B,C) and their values with lower-case letters (a,b,c). If A = a we say that A has been instantiated.A set of variables is denoted by a bold upper-case letter (**X**), and a particular instantiation by a bold lower-case letter (**x**). For example if **X** represents the variables A,B,C then **x** is the instantiation a,b,c. The number of variables in **X** is denoted |**X**|. The number of possible states of a discrete variable A is denoted |A|.The notation pa(X) is used to refer to the parents of X in a graph.

- We use P(A) to denote the probability of A.
- We use P(A,B) to denote the joint probability of A and B.
- We use P(A | B) to denote the conditional probability of A given B.

Probability

P(A) is used to denote the probability of A. For example if A is discrete with states {True, False} then P(A) might equal [0.2, 0.8]. I.e. 20% chance of being True, 80% chance of being False.

Joint probability

A joint probability refers to the probability of more than one variable occurring together, such as the probability of A and B, denoted P(A,B).

Conditional probability

Conditional probability is the probability of a variable (or set of variables) given another variable (or set of variables), denoted P(A|B).

Distributions

Once the structure has been defined (i.e. nodes and links), a Bayesian network requires a probability distribution to be assigned to each node.Each node X in a Bayesian network requires a probability distribution P(X | pa(X)).Distributions in a Bayesian network can be learned from data, or specified manually using expert opinion.

Evidence

Things that we know (evidence) can be set on each node/variable in a Bayesian network. For example, if we know that someone is a Smoker, we can set the state of the Smoker node to True. Similarly, if a network contained continuous variables, we could set evidence such as Age = 37.5.

When evidence is set on a probability distribution we can reduce the number of variables in the distribution, as certain variables then have known values and hence are no longer variables. This process is termed Instantiation.

Instantiation

The figure below shows an example of instantiating a variable in a discrete probabilitydistribution.

P(A = False, B, C, D)

B	C	D	A = True	A = False
True	True	True	0.0036	0.0
True	True	False	0.0098	0.0
True	False	True	0.0024	0.0
True	False	False	0.0042	0.0
False	True	True	0.0256	0.0
False	True	False	0.0432	0.0
False	False	True	0.0064	0.0
False	False	False	0.0048	0.0

P(B, C, D)

C	D	B=True	B=False
True	True	0.0036	0.0256
True	False	0.0098	0.0432
False	True	0.0024	0.0064
False	False	0.0042	0.0048

Figure 2 : An example showing Instatiation

2.4.Deep Learning Techniques

Today, neural networks (NN) are the most commonly used predictive models for predicting traffic flow as they are capable of modeling non-linear, stationary / non-stationary behavior and are very scalable. This means that the property of space-time can also be taken into account and that environmental data sources can be easily integrated. All of these properties reduce the unpredictable part of the uncertainty, which could also significantly reduce the prediction error in extreme cases.

2.4.1. Artificial Neural Networks

The models of artificial neural networks (NAR) have been widely studied with the aim of obtaining human-type performances, in particular in the field of pattern recognition.

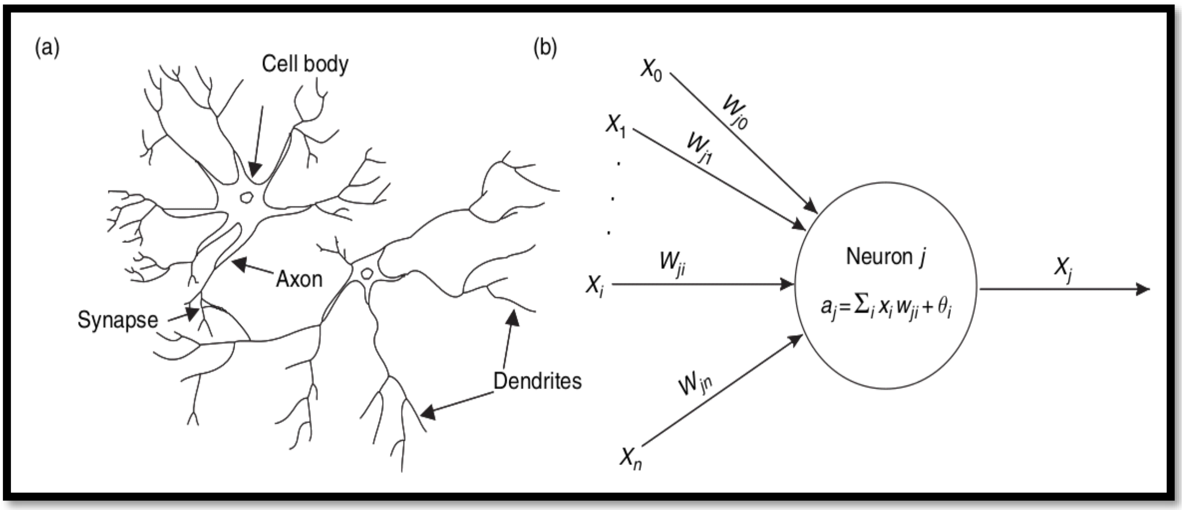
These networks are made up of a series of non-linear computing elements that work in parallel and are arranged in a way that recalls biological neural connections.

Some background work in the field of artificial neural networks (ARN) took place in the late 19th century and early 20th century. It was mainly an interdisciplinary work in physics, psychology and neurophysiology. This first work focused on general theories of learning, others, conditioning, etc., and did not contain any specific mathematical model of neural function. These new developments have revived the field of neural networks. In the past two decades, large offers of "items and many different types" of RNA have been published. Neural networks have been used in a variety of fields, including aerospace, automotive, banking, defense, electronics, entertainment, finance, insurance, manufacturing, medicine, oil and gas , voice, securities, telecommunications, transportation and the environment. In the ecological field, ANN models were used in the early 1990s, but became increasingly popular from the 1990s.

2.4.2. Biological Inspiration

A human brain is made up of approximately 10 10 neurons, computer elements that communicate via a connection network (approximately 10 4 connections per element). The years act as parallel distributed computer networks and are analogous to biological neural systems in some basic properties (Figure 13). There are many input signals (X 1⁄4 1⁄2 x 1; x 2;

...; X n #) to the neurons. Each entry receives a relative weight (W 1⁄4 1⁄2 w 1; w 2; ...; W n #) affects the impact of this entry. Weights are adaptive coefficients in the network that determine the intensity of the input signal. The neuron output signal (NET) is generated by the summing block, which roughly corresponds to the biological cell body, and adds all the weighted algebraic inputs.



Different types of Ann have been developed over the past 10-15 years, but two main categories can be easily identified depending on how the learning process is:

- **In 'supervised learning'**, there is a teacher who in the learning phase 'tells' the ANN

how well it performs or what the correct behavior would have been.

- **In 'unsupervised learning'**, the ANN autonomously analyzes the properties of the data set and learns to reflect these properties in its output.

Model name	Model type	Data models
Instantanerous Travel Time	Naive	Scalar
Historical Average	Naive	Scalar

ARIMA	Parametric	Scalar
SARIMA	Parametric	Scalar
STARIMA	Parametric	Scalar
KARIMA	Parametric	Scalar
ARIMAX	Parametric	Vector
Kalman Filter	Parametric	Scalar
Bayesian Networks	Non-Parametric	Scalar, Vector, Matrix
K-Nearest Neighbors	Non-Parametric	Vector
Feed Forward Neural Networks	Non-Parametric	Scalar
Time Delayed Neural Networks	Non-Parametric	Scalar
Recurrent Neural Networks	Non-Parametric	Scalar
Long-Short Term Recurrent Neural Networks	Non-Parametric	Scalar
Gated Recurrent Unit Neural Networks	Non-Parametric	Scalar
Convolutional Neural Networks	Non-Parametric	Scalar, Vector, Matrix
Combination of CNN and FFNN	Non-Parametric	Scalar, Vector, Matrix
Combination of CNN and LSTM	Non-Parametric	Scalar, Vector, Matrix

a. Convolutional Neural Networks (CNN):

The only notable difference between CNNs and traditional ANNs is that CNNs are mainly used in

Table 1 : A comparative list of traffic prediction models[109]

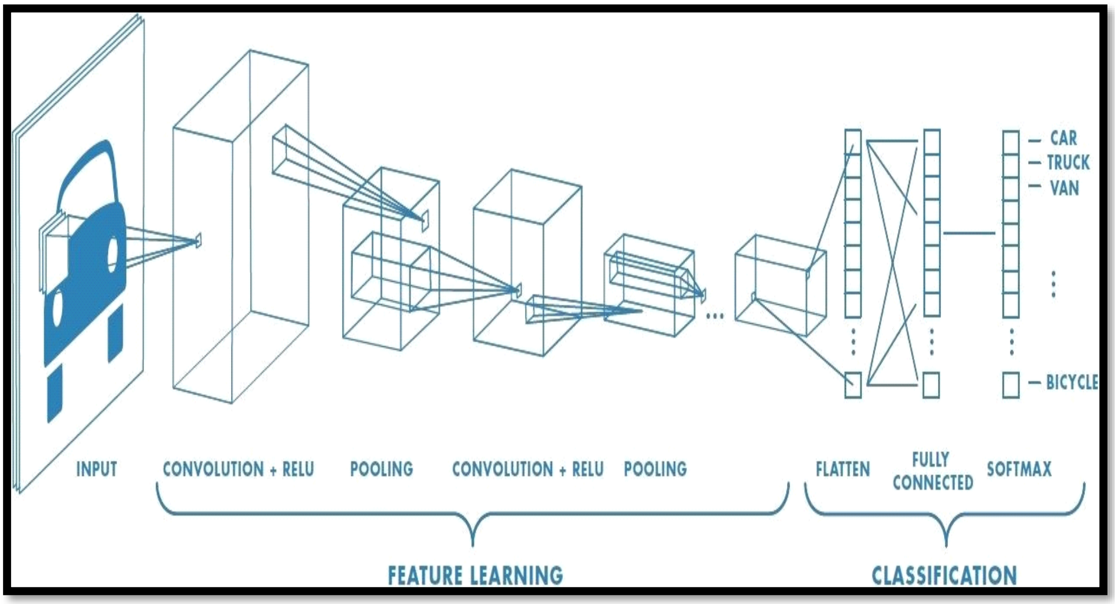
image recognition. This allows us to encode image-specific functions in the architecture, which makes the network more suitable for image-based tasks - and at the same time further reduces the parameters required to configure the model.

One of the biggest limitations of traditional ANN forms is that they face the computational complexity required to calculate image data. Common reference data sets for machine learning, such as the MNIST database for handwritten numbers, are suitable for most ANN forms due to their relatively small image size of only 28-28. With this dataset, a single neuron in the first hidden layer contains 784 weights

(28 * 28 * 1 or 1, knowing that MNIST is normalized in black and white values), which is manageable for most ANN forms.

When looking at a larger input color image from 64 to 64, the number of weights on a single neuron in the first layer increases considerably to 12,288. Also take into account that the network to process this input scale must also be much larger than the network used to classify standardized MNIST numbers in color, we will therefore understand the disadvantages of using such models.

CNNs are made up of three types of layers. These are convolutional layers, pooling layers and fully connected layers. When these layers are stacked, a CNN architecture was



formed.

Convolutional layer

As its name suggests, the convolutional layer plays an essential role in the functioning of CNNs. The layer settings focus on the use of training kernels (filters).

These grains are generally of small spatial dimension, but extend over the entire depth of the entrance. When the data reaches an entangled layer, the convolutional layer filters through the spatial dimension of the input to create a 2D activation map.

If we slide into the entry, the dot product is calculated for each value of this kernel. From there, the network learns which nucleus "triggers" when it sees a certain characteristic for a certain spatial position of the entrance. These are commonly called activations.

Figure 3: Layers of CNN

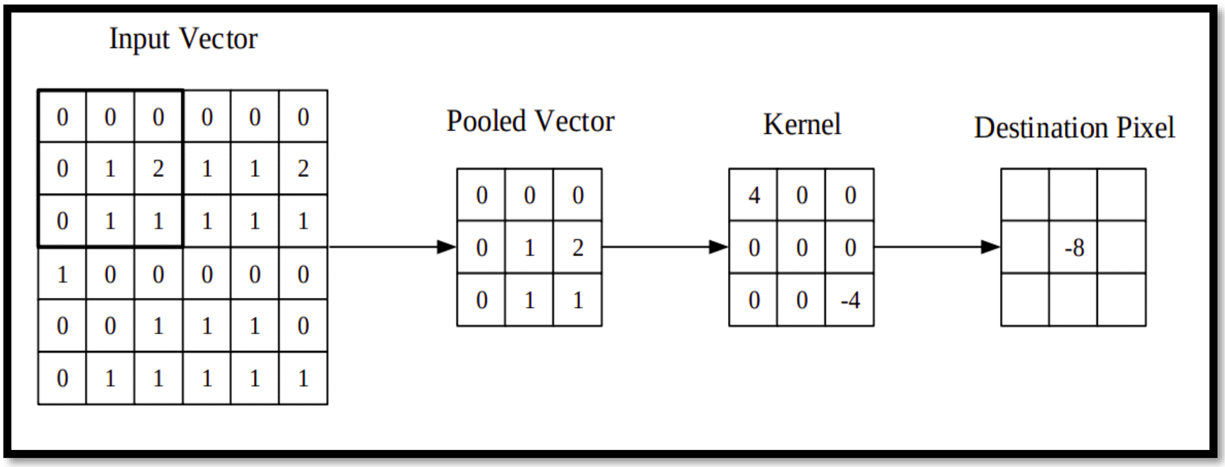


Figure 4 : Convolution operation[110]

Pooling layer

The pooling layers aim to gradually reduce the dimensionality of the representation, and therefore further reduce the number of parameters and the computational complexity of the model.

The grouping layer operates on each input activation card and scales its dimensionality using the "MAX" function. In most CNNs, these are in the form of maximum grouping layers with 2 × 2 dimension kernels applied with a stride of 2 along the spatial dimensions of the entrance. This reduces the activation card to 25% of the original size - while keeping the depth volume at its standard size.

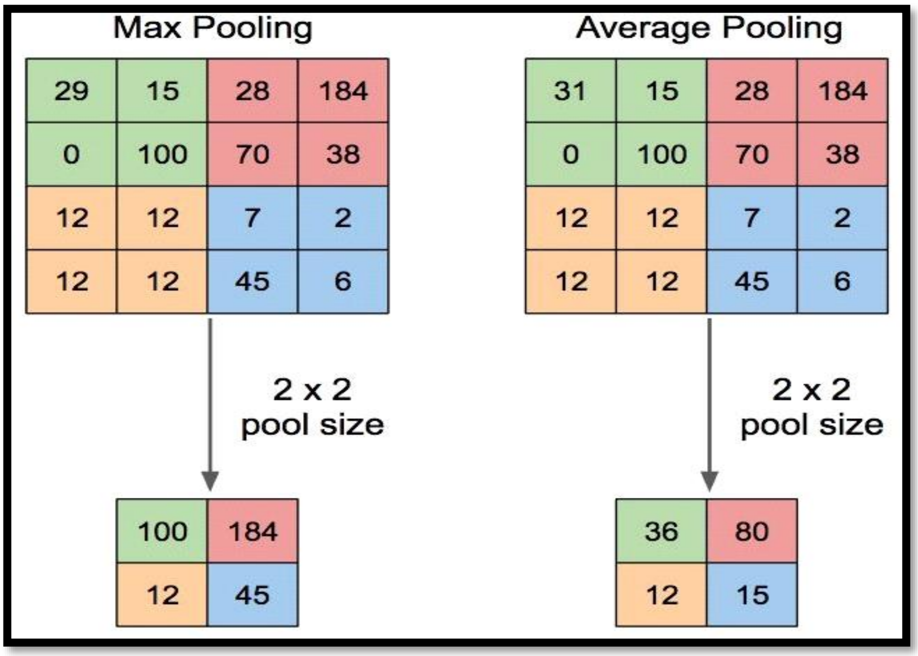


Figure 5 : Average and max pooling kernels

Fully-connected layer

The fully connected layer contains neurons which are directly connected to the neurons in the two adjacent layers without being connected to a layer in them. This corresponds to the way neurons are arranged in traditional ANN forms.

b. Recurrent Neural Networks (RNN)

Recurrent neural networks (RNN) are a type of neural network in which the output of the previous floor is introduced into the entry of the current floor. In traditional neural networks, all inputs and outputs are independent of each other, but in cases where it is necessary to predict the next word in a sentence, the previous words are necessary and therefore it is necessary to memorize them. This is how RNN was created, which solved this problem using a hidden layer. The most important feature of RNN is The Hidden State, which stores certain information about a sequence.

RNNs have a "memory" that remembers all the information about what has been calculated. It uses the same parameters for each input, it performs the same task on all the hidden inputs or layers to produce the output. This reduces the complexity of the parameters, unlike other neural networks.

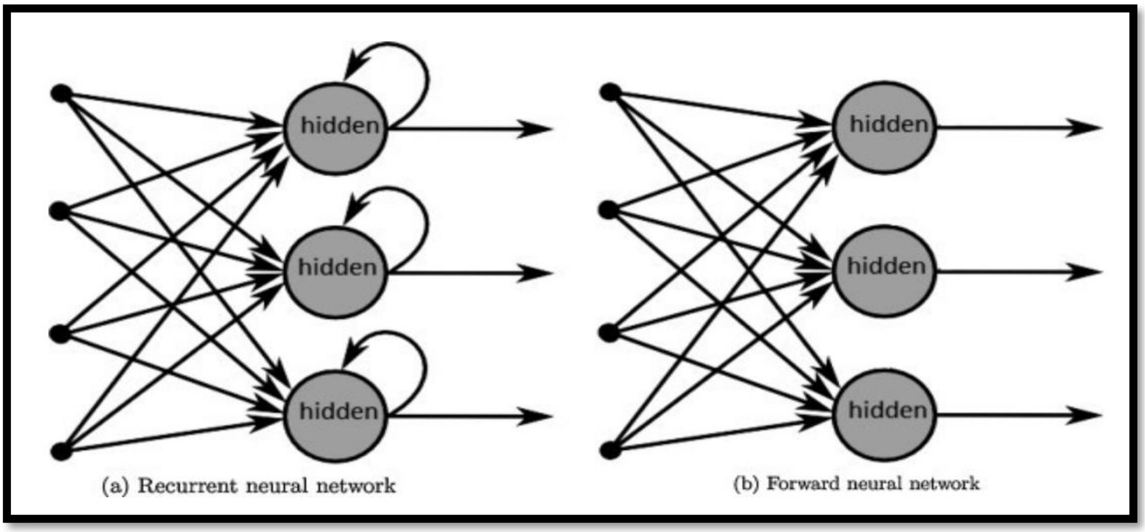


Figure 6 : Types of recurrent neural networks

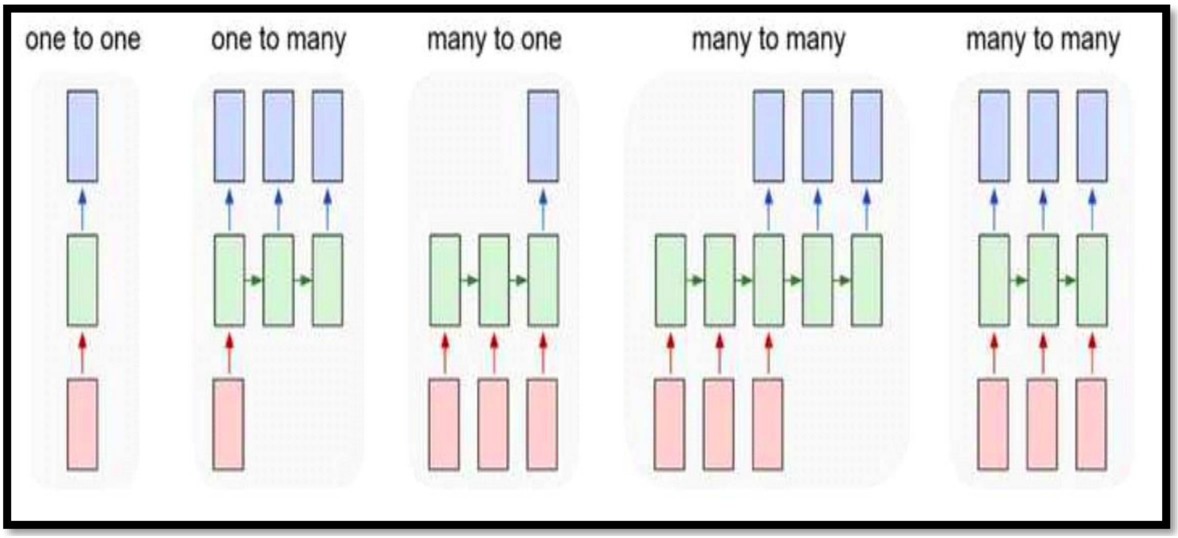


Figure 7 : the difference between forward neural networks ans Recurrent neural networks

Conclusion

We talked more about the practical side in this chapter, we started from the standard methods used in traffic flow forecasting with examples, and by the end of the chapter we explained two of the most used deep learning algorithms. We also talked about the literature and the advancement of techniques used in this field.

The next chapter will be the practical side of our work, it will contain a detailed review of our two phase work, and the results of the deep learning models implemented in Python learning language

CHAPTER 03

CNN Deep Learning Architecture Based On Traffic Simulation Model For Traffic Flow Pediction

1. Traffic Prediction Problem
2. Our aim
3. A deep Learning
4. Programing Tools
5. Scenario Description

Introduction

The purpose of this chapter is to introduce the research methodology for predicting the state of traffic using deep learning sequential model, and comparing it with a CNN model, to obtain a deep understanding of the advantages and disadvantages of proposed architectures and the difficulties of the previously mentioned problem. In this chapter we will also describe the steps we followed to achieve a satisfying result to our aimed objective.

In our work we have applied a sequential model, implemented using TensorFlow and Keras libraries, exploiting the features of these packages helped us achieve better results with a reduced source code, before that we extracted a dataset from a simulated scenario created with SUMO.

The applicability of the approach is discussed in depth in this chapter. The research plan, including the methodology, deep learning difficulties for predicting traffic congestion, the proposed method, the scenario and the software applications used to apply it, and conclusively we will evaluate the suggested models.

1. Traffic prediction problem

Artificial intelligence has been a major reason for pushing the boundaries of many areas and achieving much better results. Our field of study, urban traffic forecasting, has improved remarkably over the years by applying different machine learning and deep learning models. Many problems and difficulties have been solved, and many more have appeared, each approach was a step forward and helped to achieve better performance. In the following, we present main issue in traffic flow prediction:

- Sharp changes in flow: Traffic forecasting becomes difficult due to short-term (for example, accidents, construction) and long-term (for example, peak, seasonal, weather) traffic patterns. While most of the proposed techniques focus on forecasting normal conditions, a significant number of deep learning approaches do not include forecasting traffic under extreme conditions.
- Non-linearity of data: Traffic flows are non-linear, mostly non-stationary processes, influenced by many factors such as weather, day of the week, unforeseen events, road construction and lighting conditions.
- Drivers' Behaviors: urban traffic is influenced by many factors; most of them can be predicted because of their linearity while some other variables are almost impossible to predict. In heavy rain, pedestrians try to move under covered paths like underground passages and avoid open spaces, which can lead to overcrowded areas. Motorists always drive slower when there are slippery roads and poor visual conditions, which increases the risk of accidents, which increases the likelihood and volume of traffic jams and congestion.

2. Our aim

In order to build a model capable of predicting the future state of traffic flows with great precision, we had to divide our work into two phases, simulation and implementation.

- The first phase was necessary due to the lack of data we needed to train our deep learning model, so we had to simulate a scenario using SUMO
- In the second phase, we implemented two models in Python programming language.

3. A deep Learning Architecture For traffic flow prediction

In this part, we created two deep learning models, but before implementing them, we had to process the data in a preprocessing phase, the application of these preprocessing procedures helped us to improve the capacities. of our models.

3.1. Pretreatment phase

The data were extracted from a simulated scenario, it is a matrix containing 4 columns and the number of rows differs according to the data set (training data set or test data set). The input data was the first three columns: time, density and speed, while the fourth column Traffic-Flow is the output data.

The time column was in time format, ie "12:00:00 AM", we want all input data in float type so we changed the format of the column, we first converted it to 24h format, ie "04: 00: 00 PM" becomes "16:00:00", after which we deleted the ":", therefore it becomes " 160000 ", the process was applied using the function time(t).

The "Traffic-Flow" output column is the speed at which the vehicles pass the simulated point studied (vehicles per hour), its state can be: free traffic, congestion or congestion, and to demonstrate these states, we have applied the function flow (f), this function returns 0 when the traffic flow is between 0 and 1200, returns 1 when the traffic flow is between 1200 and 2200 and returns 2 if the value of the traffic flow is greater than 2200, after what we applied the to_categorical () method from Keras Library to transform our column into a trainable model.

This preprocessing process was sufficient to increase the accuracy of the first model to its highest values, but for the second model, additional procedures were applied. For the second model, we had to convert the values of the input data to make them between -1 and 1 using the following formula:

$$\frac{X-\bar{X}}{Xmax-Xmin} \quad (23)$$

Where X is the actual, is the average of the column's values, Xmax is maximum value in the columns where Xmin is the minimum value of the column.

We applied this formula on each column of the input data, by the following functions: **convert_speed(), convert_density()** and **convert_time()**. Because of using convolutional layer as an input layer for the second model, input data had to be converted from a matrix to a vector.

3.2. Proposed ANN model for prediction

The ANN model that we propose is based on fully connected layers to learn automatically and in a supervised manner the spatio-temporal characteristics for the classification of data extracted from the simulator. We will first describe the architecture, then we are interested in learning by playing with the parameters.

3.2.1. Architecture

The architecture proposed in the following figure on the data set generated by the SUMO simulator in the previous section. This example has 4 layers: an input layer, an output layer and 2 hidden layers.

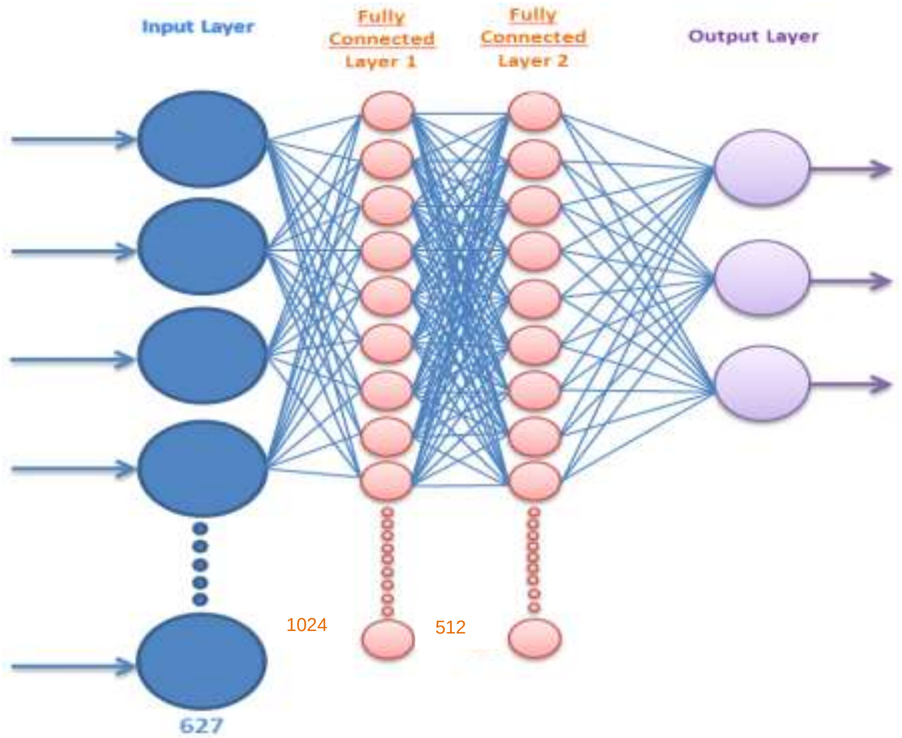


Figure 1 Proposed architecture of ANN

The input layer takes a matrix containing 3 columns and 6898 rows, the columns are time, density and speed. this input data has been processed, as mentioned in the preprocessing phase.

The output layer contains 3 neurons which represent the different traffic states, while the hidden layers are all of the same category which is fully connected.

- **First fully connected layer**: The number of neurons in this layer is the number of rows of the dataset, the layer with 1024 units and a Rectified Linear Unit (ReLU) as an activation function, and input_dim = 3 (the number of columns in the input layer).

- **Second connected layer**: The second fully connected layer has 512 units with ReLU as an activation function, which are the parameters as the first layer.

- **Output layer**: is only with 3 neurons that represents the number of possible states, in this layer we applied a Softmax activation function.

3.2.2. Learning phase

Our classification model with this architecture (ANN), embodied with 530000 trainable parameters; and a number of samples from 627 before the increase in our data set. The increase in data improves the accuracy of the ANN and reduces over-adjustment [17]. To reduce over-fitting, we increase the number of rows in the dataset matrix.
This Configuration Model (the parameters that are used in our architecture for learning) is as follows:

- **Model compilation:** we are going to apply these parameters for compiling our models:

  - ✓ **Compile ():** Compile defines the loss function, the optimizer and the metrics

  - ✓ **Adam Optimizer ():** We will use the Adam optimization algorithm to update the iterative network weights that are based on training data.

  - ✓ **loss = 'categorical_crossentropy' ():** we use this loss function for categorizing a label.

  - ✓ **Metrics = [‘accuracy’]:** we use this function to evaluate the performance of our model.

- **Training the model:** in the learning phase we use these parameters

  - ✓ **fit_generator ():** used to train our deep learning models (ANN). It requires a generator for training data

- ✓ **Batch size:** (64): the batch size used in our ANN model is 64 because it is a good starting point.

- ✓ **Epochs:** using the epochs to separate the training into separate phases, which is useful for journaling and periodic evaluation. We respectively use 50,60,70,100,150 numbers of epochs to obtain perfect accuracy for our model.

- **Testing the data**:

To evaluate the approach, we performed experiments on a set of data in a variety of urban traffic states. The test data composed of 6898 rows of data. This scenario is to obtain an accuracy of 81% in the evaluation of our ANN model before the increase of our dataset. We have applied the same techniques on preprocessed data which explains the steps that we applied on the dataset.

3.2.3. Interpretation of the proposed model

The accuracy of learning and testing increases with the number of epochs, which reflects that each time the model learns more information. As seen in the fig above.

Figure 20 shows changes in the accuracy of training where:

- From 0 epoch to 60 epochs, the precision of the training increases up to the value 80% at epoch 60.

- The maximum training accuracy reaches 80%.

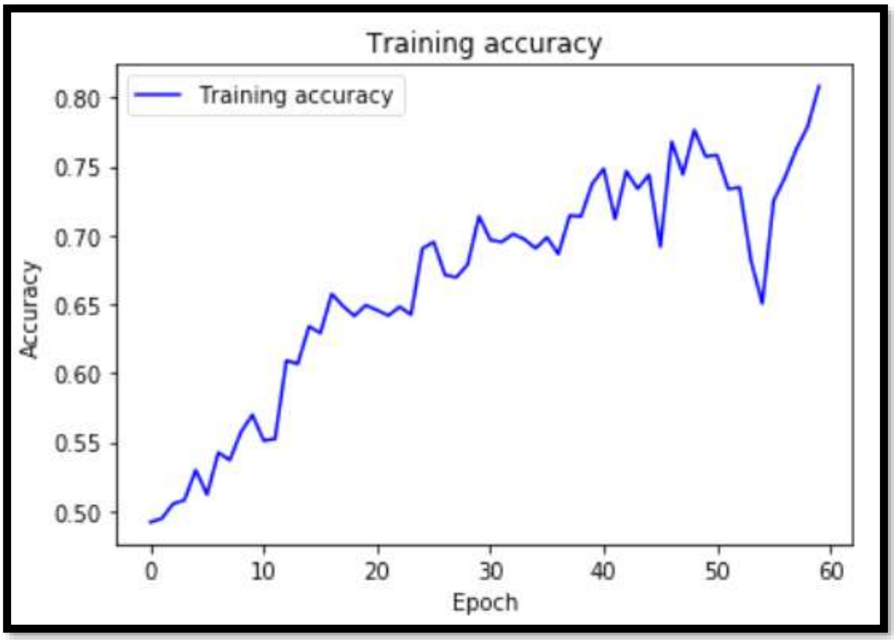


Figure 2 : model accuracy

Figure 21 shows changes in The Training Error where:

- From 0 epoch to 60 epochs, the loss accuracy decreases from the value 3500 to the value close to 0.5004 at epoch 60.
- The minimum training loss is 0.5004.

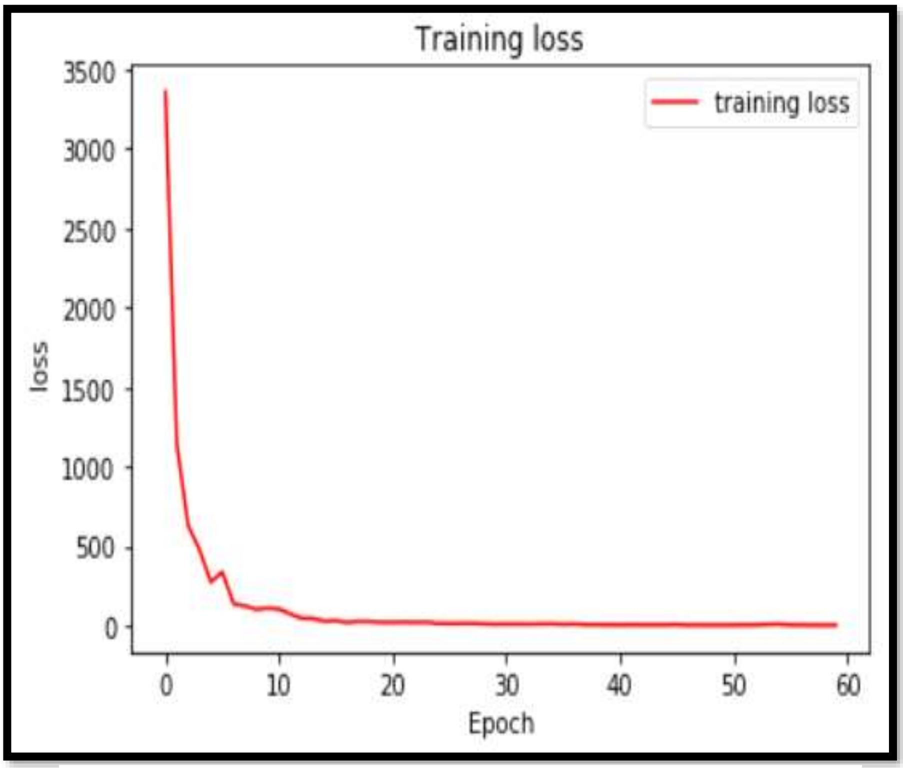


Figure 3 : model training loss

And here is the precision and the learning loss in the following figure:

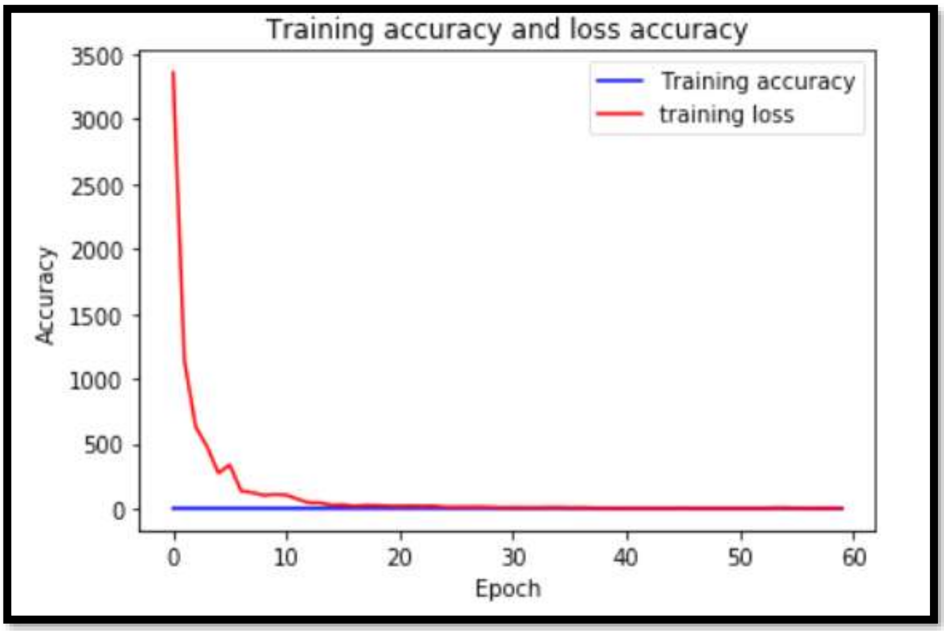


Figure 4 : model accuracy and loss

3.3. Proposed CNN model for prediction

The CNN model that we propose is based on layers of convolution layers to learn automatically and in a supervised manner the spatio-temporal characteristics for the classification of data extracted from the simulator. We will first describe the architecture, and then we are interested in learning by playing with the parameters.

3.3.1. Architecture

The architecture proposed in the next figure on the dataset generated by SUMO simulator in the previous section. This example has 4 layers: An input layer, an output layer and 2 hidden layers.

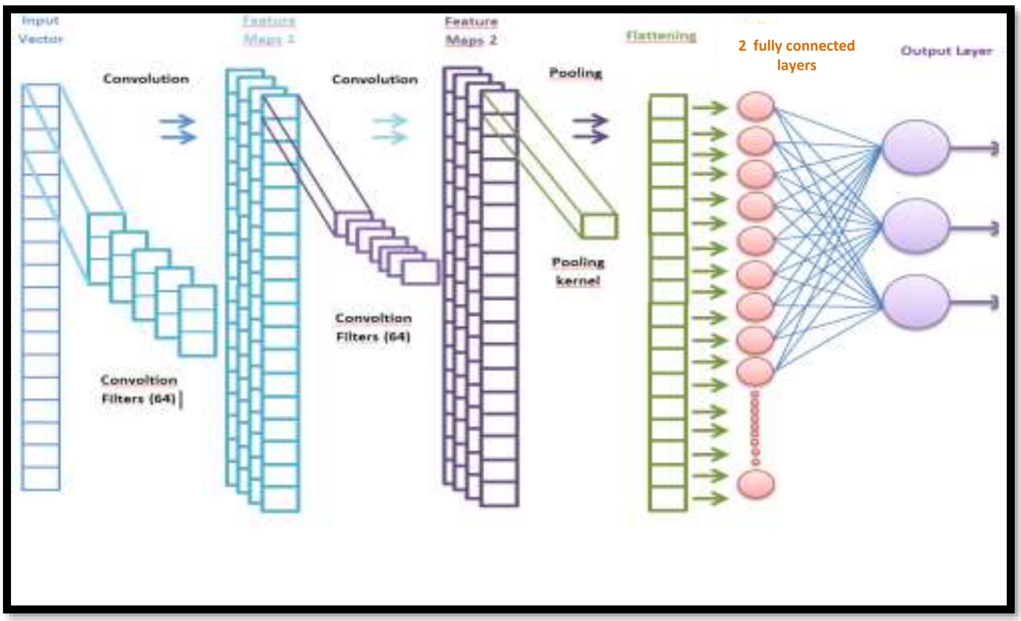


Figure 5 : Proposed architecture of CNN

The input layer takes a matrix containing 3 columns and 6898 rows, the columns are time, density and speed. this input data has been processed, as mentioned in the preprocessing phase.

The output layer contains 3 neurons which represent the different traffic states, while the hidden layers are all of the same category which is fully connected.

- **First Convolution Layer**: the number of filters is 64,
- **Second Convolution Layer**: The second fully connected layer has 50 units with ReLU as an activation function, which are the parameters as the first layer.
- **Max Pooling Layer**: pool size of 3, with 2 stride and the padding is set to 'same'.
- **The First Fully Connected Layer**: 64 units and ReLU activation function
- **The second Fully Connected Layer**: 32 units and ReLU activation function

3.3.2. Learning phase

Our classification model with this architecture (CNN), embodied with 530000 trainable parameters; and a number of samples from 627 before the increase in our dataset. The increase in data improves the accuracy of CNN and reduces overfitting [17]. To reduce over-fitting, we increase the number of rows in the dataset matrix.

This Configuration Model (the parameters that are used in our architecture for learning) is as follows:

- **Model compilation :** we are going to apply these parameters for compiling our models:
  - ✓ **Compile ():** Compile defines the loss function, the optimizer and the metrics
  - ✓ **Adam Optimizer ():** We will use the Adam optimization algorithm to update the iterative network weights that are based on training data.
  - ✓ **loss = 'categorical_crossentropy' ():** we use this loss function for categorizing a label.
  - ✓ **Metrics = ['accuracy']:** we use this function to evaluate the performance of our model.
- **Training the model:** in the learning phase we use these parameters

- ✓ **fit_generator ():** used to train our deep learning models (CNN). It requires a generator for training data.
- ✓ **Batch size:** (64): the batch size used in our CNN model is 64 because it is a good starting point.
- ✓ **Epochs:** using the epochs to separate the training into separate phases, which is useful for journaling and periodic evaluation. We respectively use 50,100,150 number of epochs to obtain perfect accuracy for our model.

3.3.3. Testing the data:

To evaluate the approach, we performed experiments on a set of data in a variety of urban traffic states. The test data (test data) composed of 627 rows of data. This scenario is to obtain an accuracy of 50% in the evaluation of our ANN model before the increase of our dataset. We have applied the same techniques on preprocessed data which explains the steps that we applied on the dataset.

4. Programming Tools

The study was conducted using various tools in both phases (simulation and implementation)

4.1. Simulation of Urban Mobility (SUMO)

SUMO is an open source road traffic simulation software package designed to manage large road networks. It models intermodal traffic systems, including road vehicles, public transport and pedestrians. It covers a multitude of support tools, which manage tasks such as route searching, viewing, importing networks and calculating emissions. SUMO can be improved with custom models and provides various APIs to remotely control the simulation.

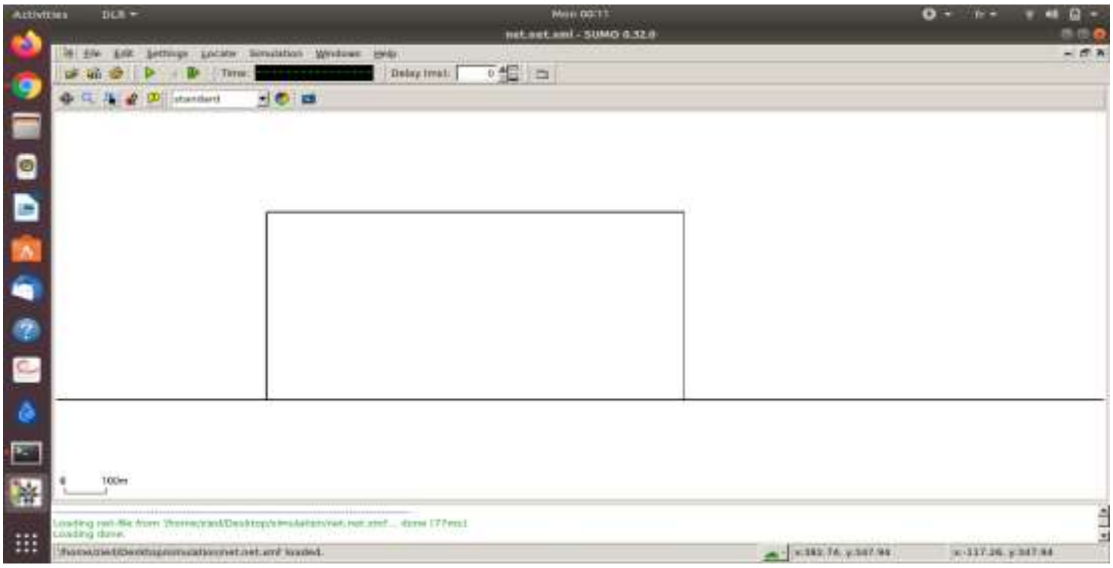


Figure 6: a screenshot of SUMO-GUI

Some SUMO features:

- Microscopic simulation
- Online interaction
- Multimodal simulation incl. vehicles, public transport, pedestrians
- Automatic generation of time schedules of traffic lights
- No limitations in network size and number of simulated vehicles
- Evaluation of eco-aware routing based on pollutant emission and investigations of autonomous route choice on the overall network.

Using SUMO, we could build the dataset that we trained the deep learning model with.

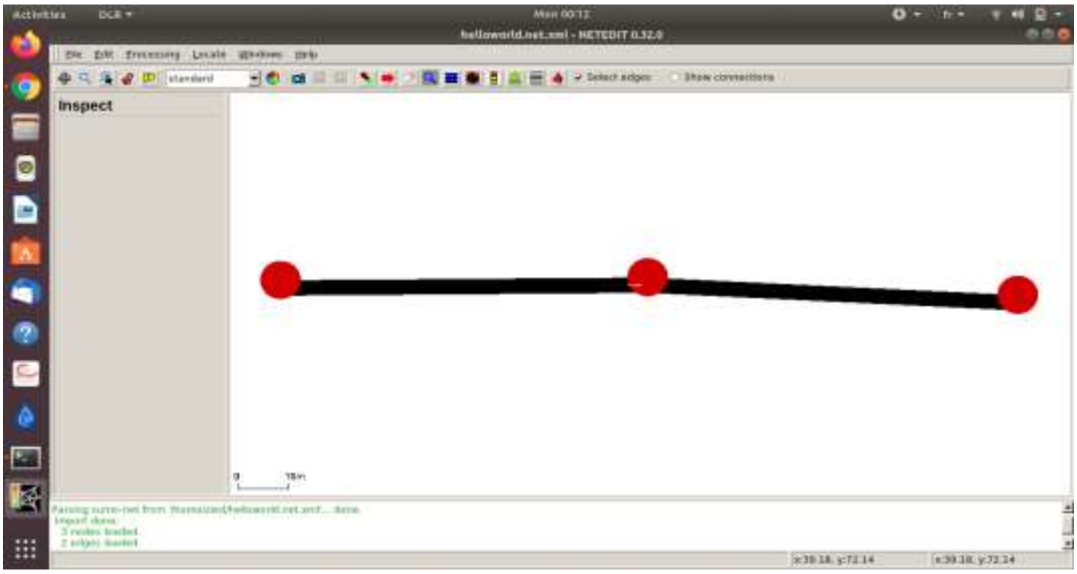


Figure 7 : a screenshot of NETEDIT

4.2. OMNeT++(Objective Modular Network Testbed in C++)

It is an extensible, modular, component-based C++ simulation library and framework which also includes an integrated development and a graphical runtime environment. Domain-specific functionality (support for simulation of communication networks, queuing networks, performance evaluation, etc.) is provided by model frameworks, developed as independent projects. There are extensions for real-time simulation, network emulation, support for alternative programming languages (Java, C#), database integration, SystemC integration, HLA and several other functions.

4.3. Veins

Veins is an open source framework for running vehicular network simulations. It is based on two well-established simulators: OMNeT++, an event-based network simulator, and SUMO, a road traffic simulator. It extends these to offer a comprehensive suite of models for simulation.

4.4.Python

Python is an interpreted, high-level, general-purpose programming language. It supports several programming paradigms, including structured (in particular, procedural), object-oriented and functional programming.

Python is supposed to be an easily readable language. Its formatting is visually refined, and it often uses English keywords where other languages use punctuation. Unlike many other languages, it does not use square brackets to delimit blocks and semicolons after the instructions are optional.

In Python we imported some libraries that made the implementation of our model easier:

4.4.1. NumPy

NumPy is an arary-processing package that provides a high-performance multidimensional array object, and tools for working with these arrays, It contains various features:

- A powerful N-dimensional array object
- Sophisticated (broadcasting) functions
- Tools for integrating C/C++ and Fortran code
- Useful linear algebra, Fourier transform, and random number capabilities[115]

4.4.2. Pandas

Pandas is a package written for data manipulation and analysis, it offers data structures and operations for manipulating numerical tables and time series.

Library features:

- DataFrame object for data manipulation with integrated indexing.
- Tools for reading and writing data between in-memory data structures and different file formats.
- Data alignment and integrated handling of missing data.
- Reshaping and pivoting of data sets.
- Label-based slicing, fancy indexing, and subsetting of large data sets.
- Data structure column insertion and deletion.
- Data set merging and joining.

4.4.3. TensorFlow

TensorFlow is an open source artificial intelligence library, using data flow graphs to build models. It allows developers to create large-scale neural networks with many layers. This package is mainly used for: Classification, Perception, Understanding, Discovering, Prediction and Creation.

4.4.4. Keras

Keras is an open-source, user-friendly, modular, and extensible neural-network library that’s capable of running on top of TensorFlow, it’s designed to enable fast experimentation with deep neural networks. Keras contains numerous implementations of commonly used neural-network building blocks such as layers, objectives, activation functions, optimizers, and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural network code.

4.5. Spyder

Spyder is an open source cross-platform integrated development environment (IDE) for scientific programming in the Python language.

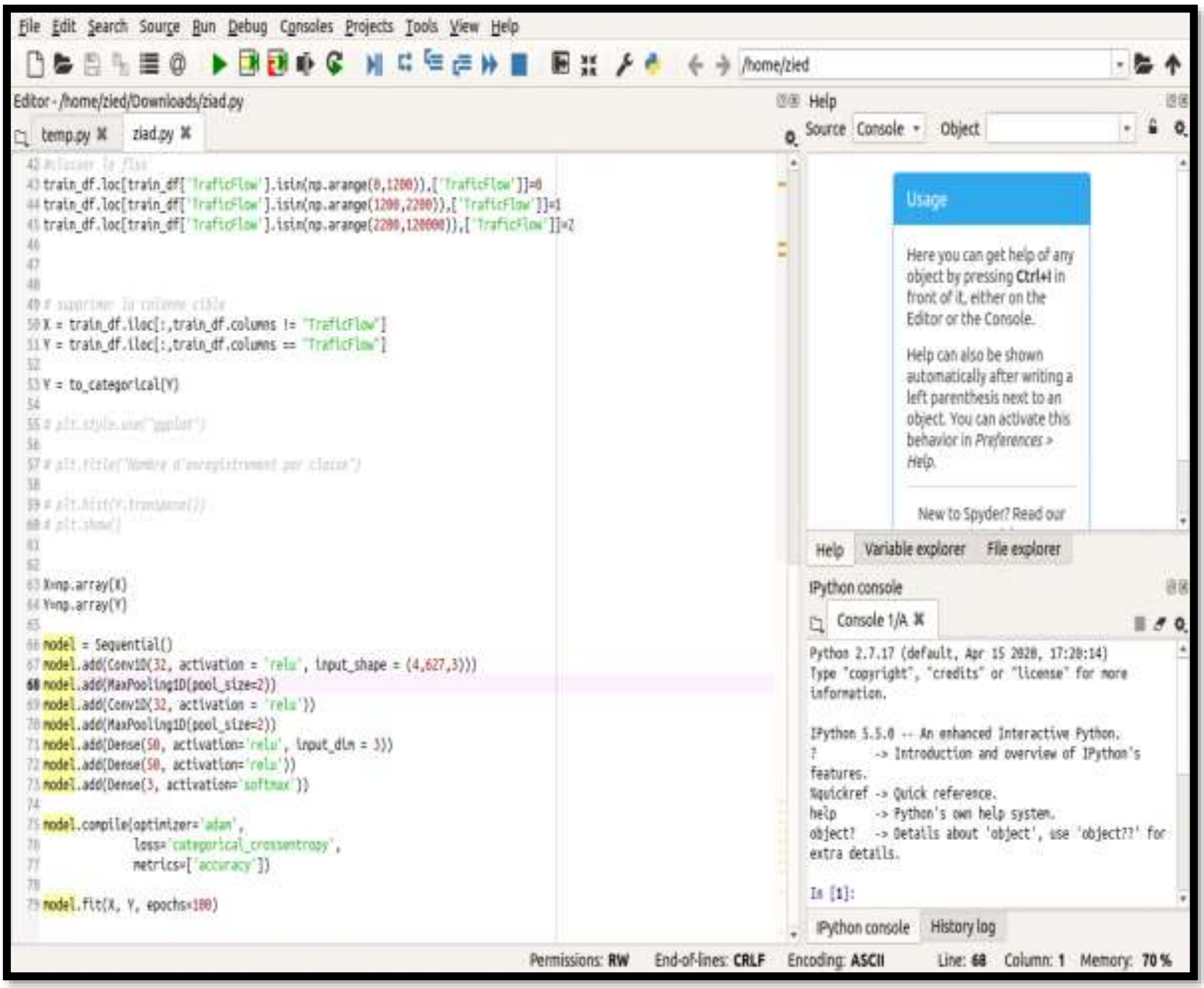


Figure 8 : a screenshot of Spyder graphical user interface

5. Scenario description

To attain our objective which is the creation of a simulated scenario for traffic flow, we need some elements:

- Data of the network
- Traffic demand
- Additional traffic infrastructure

We chose to simulate a scenario because of the unavailability of real-world traffic datasets that are convenient to our case, and it’s a time-consuming process to prepare a simulation scenario based on real-world data.

Edges are an essential element in SUMO, their representation varies, it can be: street, waterways, tracks, bike lanes and walkways. They are unidirectional with one ore lanes running in parallel, their geometry is described by a series of line segments. Attributes such as width, speed limit and access permissions (for example, bus permissions only) are modeled as a constant along a track. As a result, the distance should be modeled as an edge sequence if any of these attributes change along its length. SUMO networks contain detailed information on possible movements at intersections and the corresponding rules of conduct used to determine dynamic simulation behavior. To ensure consistent network representation, SUMO networks are created using the NETCONVERT and NETEDIT applications. NETCONVERT is a command line tool for importing road networks from different data sources.

OpenStreetMap (OSM), Open DRIVE, Shapefile or other simulators such as MATSim and Vissim. An essential feature of NETCONVERT is the heuristic refinement of missing network data in order to achieve the level of detail required for microscopic simulation (for example, synthesis of traffic light plans, rules of conduct and reduction of geometry for OSM networks).

NETEDIT is a graphical network editor that can be used to create, analyze and modify network files. This complements network generation heuristics with manual refinements and also supports the definition of additional transport infrastructure that could not be imported by NETCONVERT. Support features include:

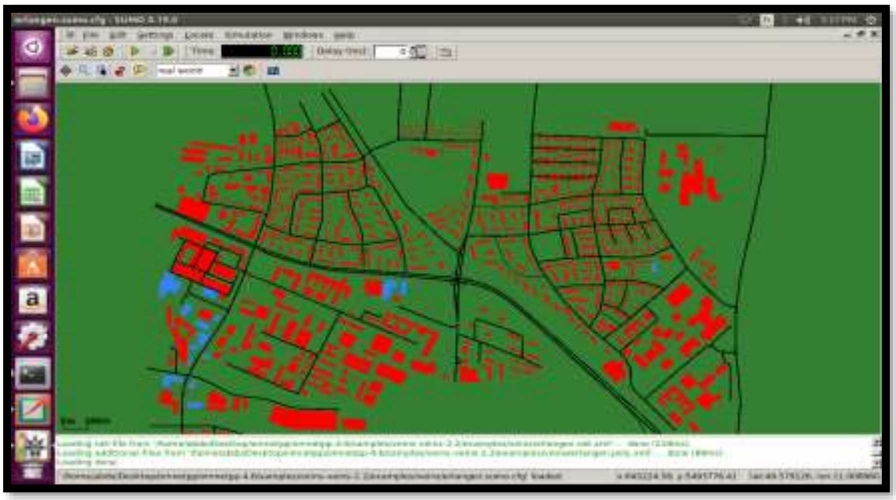
- Basic network elements
- Advanced network elements
- Polygons and points of interest

Due to the frequent discrepancy between the available input data and the level of detail required for microscopic simulation, preparing the network and infrastructure is often a demanding task. For this reason, NETCONVERT is constantly evolving to improve its heuristics and reduce the amount of annual processing required. In recent years, there have been a number of improvements in terms of:

- Modeling networks with left-hand traffic
- the creation of multimodal networks for vehicles, wheels and pedestrians
- Importing new data formats

All output files written by SUMO are in XML-format by default. A SUMO network file describes the traffic-related part of a map, the roads and intersections of the simulated vehicles. On an approximate scale, a SUMO network is a directed graph. The nodes commonly called intersections in the sumo context represent intersections and "edge" routes, the SUMO network contains the following information:

- Every street (edge) as a collection of lanes, including the position, shape and speed limit of every lane,
- Traffic light logics referenced by junctions,
- Junctions, including their right of way regulation,
- Connections between lanes at junctions (nodes).



5.1. Study scenario

In this analytical study, a supplied empirical traffic model was adopted in order to analyze it. This road segment sometimes suffers from traffic congestion, especially in the last hours of the day. In general, traffic congestion comes from the unexpected occurrence of incidents such as accidents or slowing down of vehicle speed.

We were interested in assessing the real state of traffic in this segment. The model provided is regenerated using an urban mobility simulator, called SUMO. The new traffic scenario includes a set of inductive loops to collect data between the start point and the end point of this segment.

Figure9 : the studied road segment opened in SUMO

we followed the following steps to create the scenario in SUMO simulator:

Figure 11 : Step2 of creating the scenario in SUMO

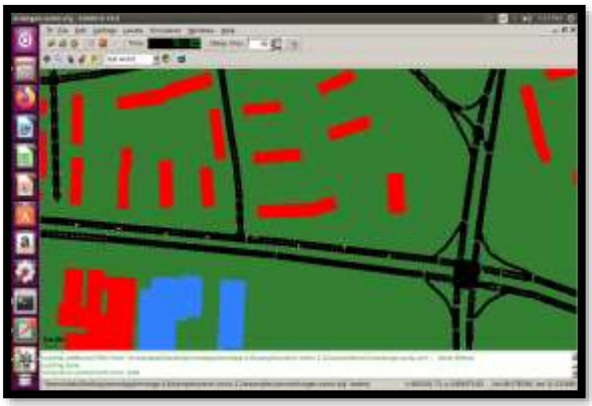


Figure 10 : Step1 of creating the scenario in SUMO

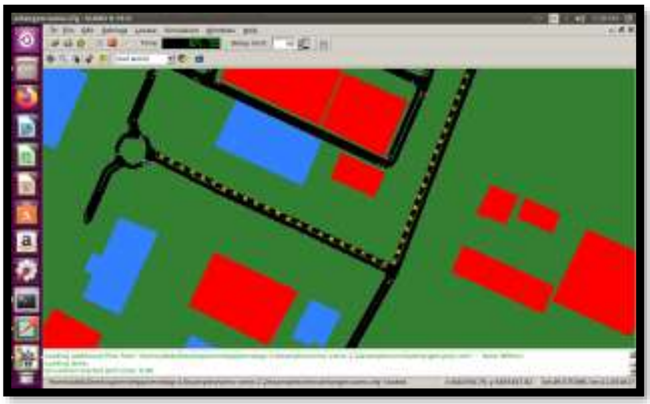


Figure 12 : Step 3 of creating the scenario in SUMO

5.2. Traffic variables model

In this section, we will present adopted traffic variables model, the following figures show the changes off trraffic variables with time, the studied variables are: speed, density and traffic flow

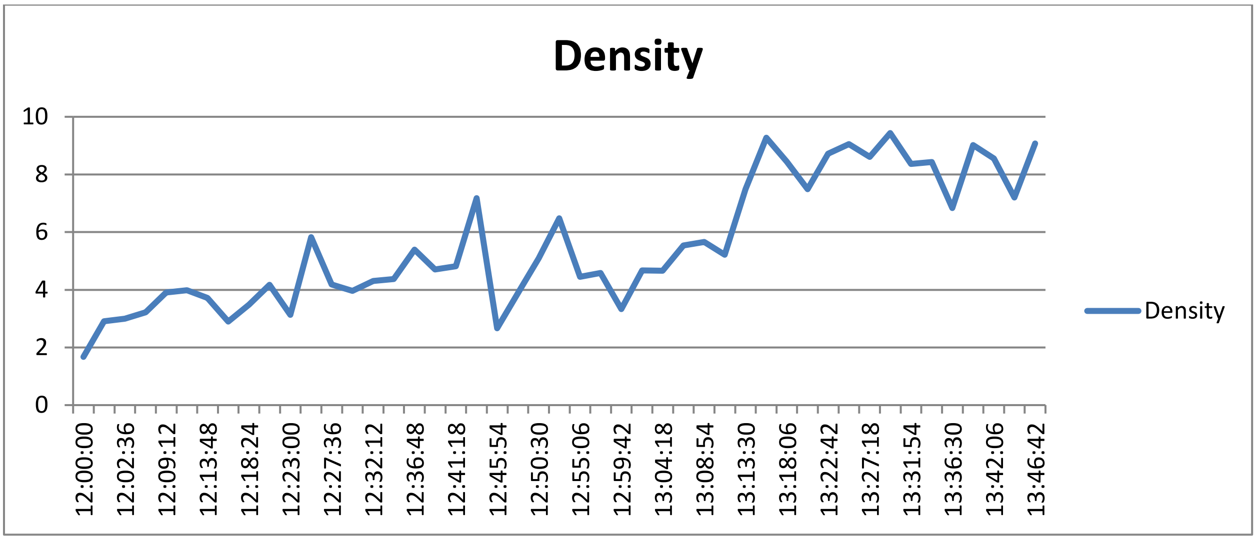


Figure 13 : graph showing the evolution of density with time

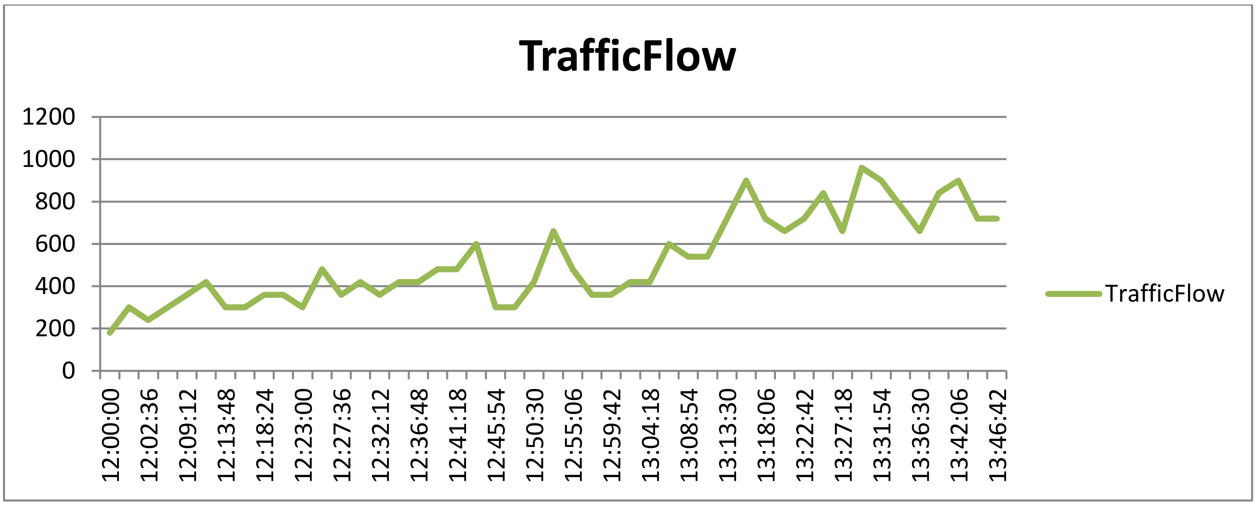


Figure 14: graph showing the evolution of Traffic Flow with time

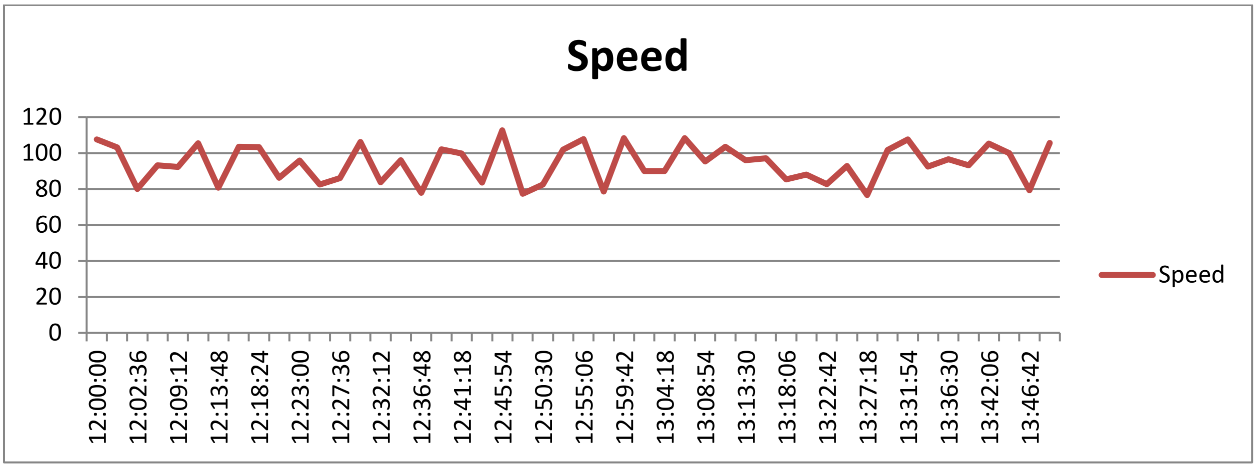


Figure 15: graph showing the evolution of Speed with time

5.3. Simulation files

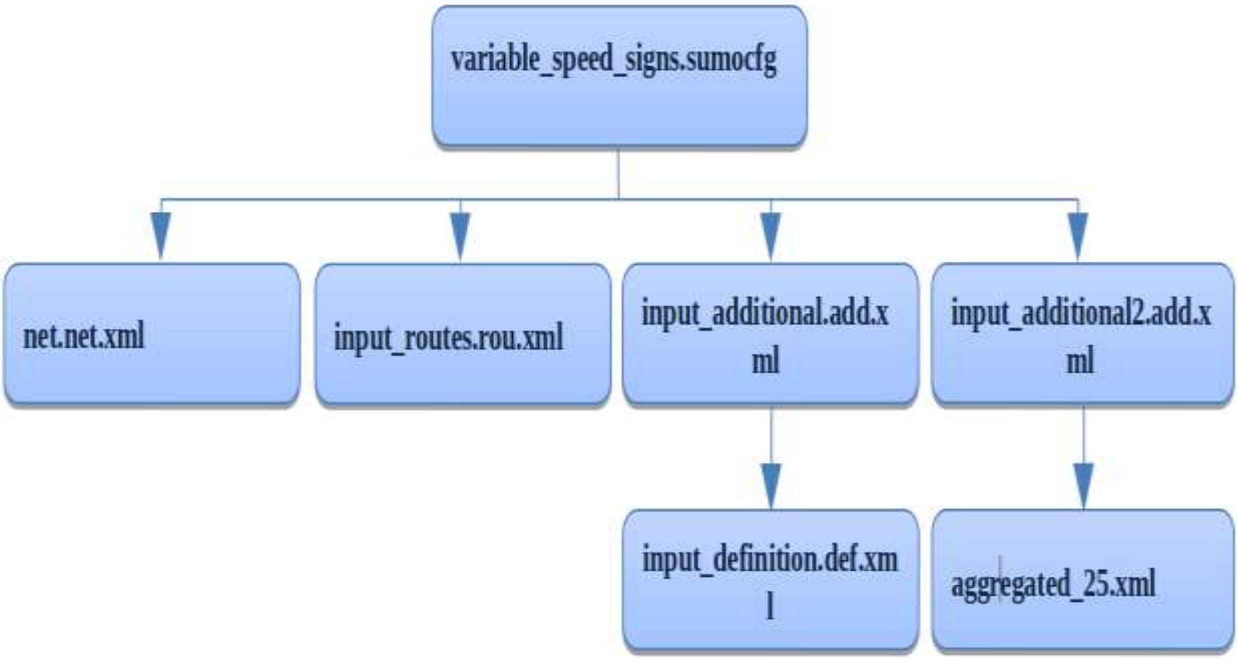


Figure 16 :A diagram showing the hierarchy of simulation files in SUMO

5.3.1. Configuration file

The main file in our scenario is « variable_speed_signs »; this file contains the input files, which are:

- Network file : `"net.net.xml"`

- Route-file : `"input_routes.rou.xml"`
- Additional-files : `"input_additional.add.xml,input_additional2.add.xml"`

The configuration file also contains the beginning and end of time values of the simulation

```
<time>
  <begin value="0"/>
  <end value="1000"/>
</time>
```

The final attribute in the configuration file is the report, in this section two variables are set :

```
<report>
  <no-duration-log value="true"/>
  <no-step-log value="true"/>
</report>
```

Figure 17 : A screenshot of configuration file

5.3.2. Network file : `net.net.xml`

It’s the first imported in the simulation document, in contains various informations about our network:

- Location
- Edges : our network is composed of 14 edges, 7 of them are internal, and the others with a priority equals -1

- Lanes: Each edge includes the definitions of lanes it consists of.

```
<edge id="<ID>" from="<FROM_NODE_ID>" to="<TO_NODE_ID>"
priority="<PRIORITY>">
<lane id="<ID>_0" index="0" speed="<SPEED>" length="<LENGTH>"
shape="0.00,495.05 248.50,495.05"/>
<lane id="<ID>_1" index="1" speed="<SPEED>" length="<LENGTH>"
shape="0.00,498.35,2.00 248.50,498.35,3.00"/>
</edge>
```

- Junctions : our model contains 7 junctions, each of them is defined by an id, a type and coordinates, in addition to other attributes such as :
- Inclanes : The ids of the lanes that end at the intersection
- The ids of the lanes within the intersection the internal junction is located within and prohibit to cross the internal junction
- Requests : they are usually declared within the Junction attributes, They describe, for each link which streams have a higher priority) and force the vehicle on link to stop.
- Connections : describe how a node's incoming and outgoing edges are connected, an example of connections is to prohibit left-turns at some junctions. In our scenario there are 14 connections.

```
<?xml version="1.0" encoding="UTF-8"?>

<!-- generated on Wed Jun 17 18:37:34 2020 by SUMO netconvert Version 0.32.0
<?xml version="1.0" encoding="UTF-8"?>

<configuration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://sumo.dlr.de/xsd/netconvertConfiguration.xsd">

    <input>
        <sumo-net-file value="net.net.xml"/>
    </input>

    <processing>
        <no-turnarounds value="true"/>
        <offset.disable-normalization value="true"/>
        <lefthand value="false"/>
        <junctions.corner-detail value="0"/>
        <rectangular-lane-cut value="false"/>
        <walkingareas value="false"/>
    </processing>
```

Figure 19 : A screenshot of network file

5.3.3. Routes file

In this file, many components can be defined, but in our scenario, just three attributes were defined:

- **Vehicles's type** : the vehicles in our scenario have a length of 3, and the minimum gap between them is 2, their maxSpeed is set to 70km/h.

<vType id="KRAUSS_DEFAULT" accel="2.6" decel="4.5" sigma="0" length="3" minGap="2" maxSpeed="70" color="1,1,0"/>

- **The route :**

<route id="0" edges="beg middle end rend"/>the vehicles

- **Flow**

<flow id="0" type="KRAUSS_DEFAULT" route="0" begin="0" end="301" period="10" departPos="0"/>trip

Figure 20 : A screenshot of route file

5.3.4. additional-files

input_additional.add.xml : this file defins just one attribute, which is variableSpeedSign, the parameters of this attribute are :

- id : The id of the Variable Speed Signal element
- lane : The id of the lanes of the simulation network
- file : optional file in which the time and speed values are defined

<additional xmlns: xsi = "http://www.w3.org/2001/XMLSchema-instance" xsi: noNamespaceSchemaLocation = "http://sumo-sim.org/xsd/additional_file.xsd">
<variableSpeedSign id = "vss" lanes = "middle_0" file = "input_definition.def.xml" />
</additional>

Figure 21: A screenshot of additional file

input_additional2.add : in this file edgedata is defined, its attributes are :

- id : The id of the detector
- file : The path to the output file. The path may be relative.
- Freq : The aggregation period the values the detector collects shall be summed up. If not given, the whole time range between begin and end is aggregated.
- excludeEmpty : If set to true, edges/lanes which were not use by a vehicle during this period will not be written; default: false. If set to "defaults" default values for travel time and emissions depending on edge length and maximum speed get printed.

5.3.5. Input definition:

Figure 22: A screenshot ofthe second additional file

« input_definition.def »: in this file, SUMO sets time and speed values of the Variable Speed Signs:

<vss>

<step time="100" speed="2.8"/>

<step time="200" speed="47.22"/>

</vss>

- time : Time in which the speed will be changed
- speed: New speed (if no value or a negative value is given, the speed will be reset to the original network value)

Figure 23 : A screenshot of input definition file

5.3.6. aggregated_25.

xml: In this file all aggregated data are collected, this data can be:

- RouteLength: average route length

- Duration: average trip duration
- WaitingTime: average time spent standing (involuntarily)
- TimeLoss: average time lost due to driving slower than desired.
- DepartDelay: average time vehicle departures were delayed due to lack of road space

Figure 24: A screenshot of aggregated file

5.4. Launch traffic data in the Veins project

We imported our sumo scenario and integrated it in OMNet, we implemented the proposed scenario using veins framework, we applied the following steps to scenario in OMNet :

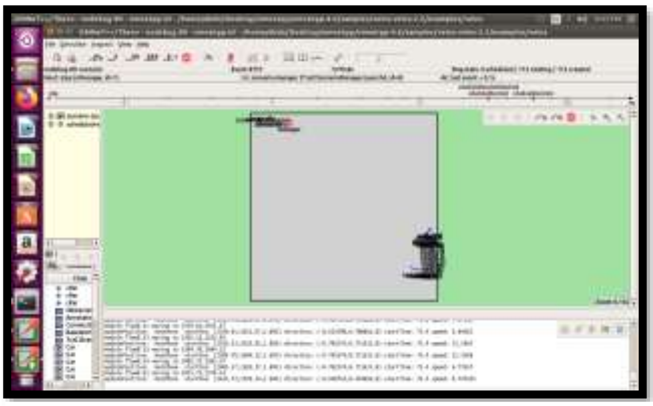


Figure 25 : Step 1 in creating a scenario in OMNet

Figure 26: Step 2 in creating a scenario in OMNet

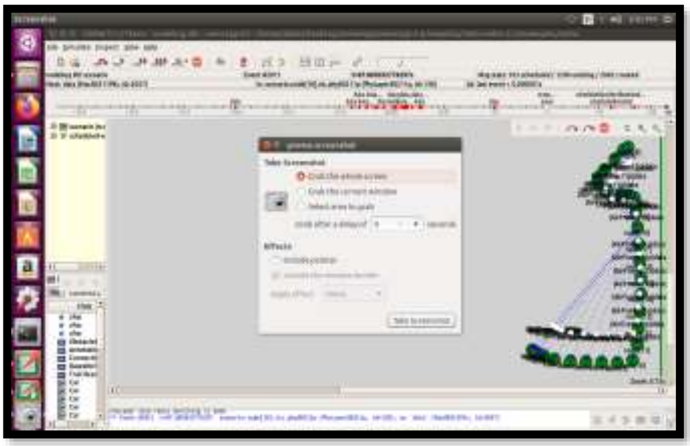


Figure 27: Step 3 in creating a scenario in OMNet

6. Model Validation

To give more concrete to our model, we opted for a validation phase, to do this we had to create a validation dataset. And for that we chose a road segment and we built a scenario using SUMO, the scenario created data which will be then used as test dataset, the aggregated data was stored in a csv file. The file composed of a matrix of 3*636, where 3 is the number of columns or our input data: Time, Density and Speed. We read the file and validated it with our learned model to attain a the traffic state of traffic in the specified situation.

The output of the prediction was a vector containing in each column another vector of size 3, the columns of the second vector contained the rate of each of the possible states, the highest value is the predicted value, for example in the first situation, the highest rate was the first which means that the traffic state of the studied road in that particular time of the day and with the extracted density and mean speed values will be in a free flow. The figure below shows the different validation steps:

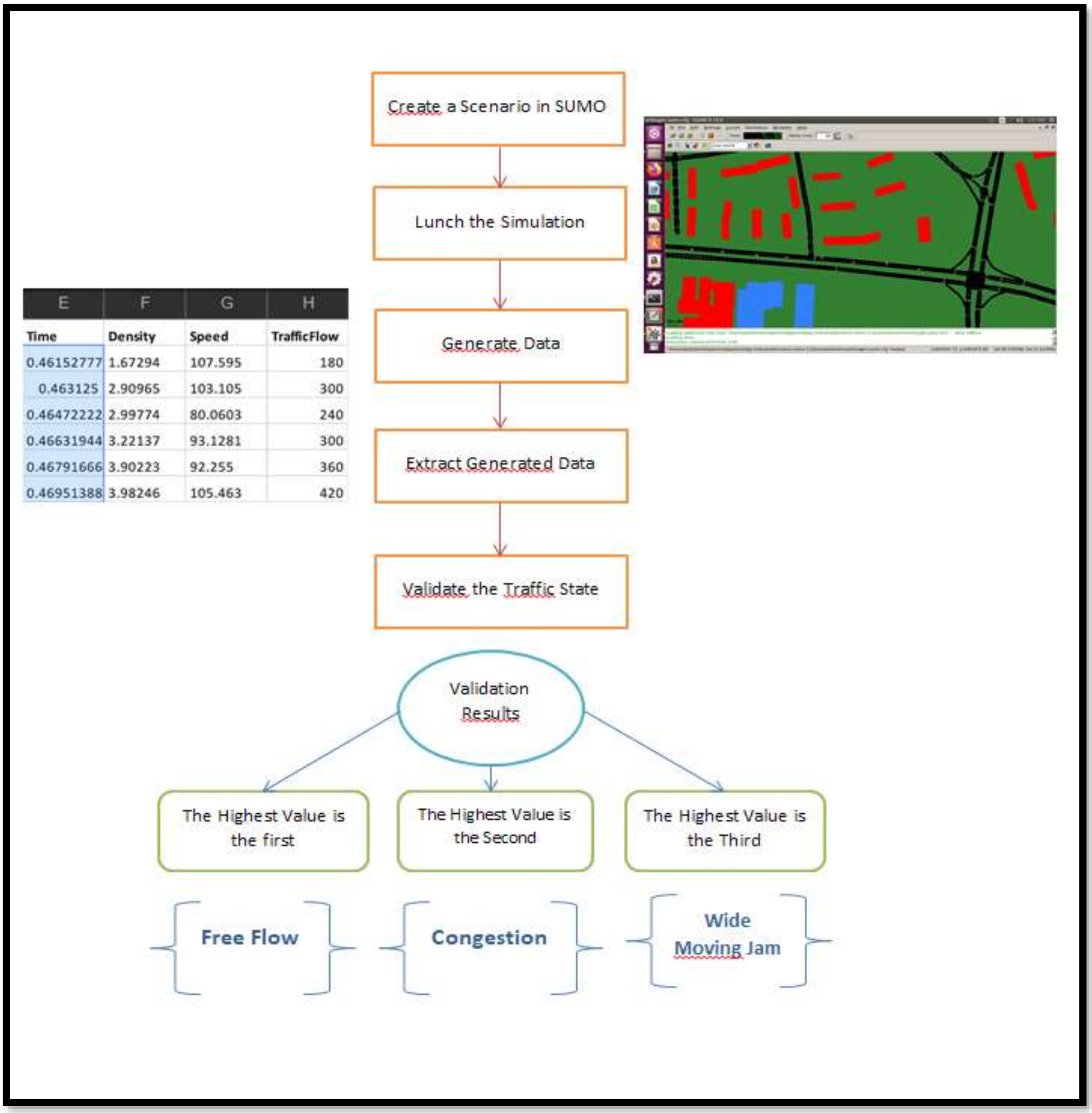


Figure 28 : Validation Model for chosen traffic stat

Conclusion

In the conclusion of the last chapter, the results we were looking to achieve were attained, with the implementation of our sequential and CNN models. The purpose of this chapter was to show all the aspects of the applicability part of our work, from the simulated scenario to the results of the proposed methods, with mentioning the used tool to achieve our objective.

The steps we followed to reach the end of our work were all shown in this chapter, in both phases.

General conclusion

We arrived at the end of our thesis with a clear vision of our problematic and satisfactory results, we conclude by recalling what our goal in writing it was. We sought to understand the problem of traffic congestion, and which methods are used to predict it, and by penetrating these techniques, we created ours and applied it with data sets extracted from a simulated scenario.

Our trip was a mixture of theoretical research and the implementation of information entered and the results of experience of researchers in the field. In the theoretical part, we started from the definition of our examined subject, automobile traffic and the problems faced by its managers, which increase over time, in this part we determined the problem that our research will focus on its congestion. After that, we thought again about the techniques commonly used to manage the traffic flow, for which we had to mention the parameters of urban traffic and their categories. Then we clarified the techniques frequently used to extract traffic data with detailed information, the types of this data were mentioned in another part. In the next part; VANETs, V2X and intelligent traffic management were the chosen applications on which we focused to analyze, at the end of this chapter we talked about performance indicators and theories of traffic flows.

The second chapter was part of the theoretical phase, and it was also an introduction to the applicable one. There, we plunged into the problem of predicting the flow of traffic, we started by evoking and the techniques which are not deep learning, to create a broader vision of our problem and its evolution, after which we spoke deep learning techniques that we have chosen to detail.

Our applicable work was divided into two parts, the first was the simulation of the scenario that created our data sets, the second was the coding part, in which we presented the models that we chose to implement. In this chapter, we have cited all of the software that we used to apply this work. We concluded this chapter by showing the results of our models, which were satisfactory.

In our work, we have tried to explore different aspects of the problem of traffic congestion, and explored the solutions proposed by the researcher, we have found that this field is very large, its problems evolve over time and become more difficult to solve with the development of the proposed solutions and their effectiveness.

In conclusion, traffic congestion is still a daily problem that has not been completely eliminated, even if its terrible effects, there are still other problems to take into account and solving them is more crucial than our discussed problem., the most serious is traffic accidents, because their costs are human lives.