Jimma University

Jimma University Institute of Technology

Faculty of Computing and Informatics

M.Sc. in Computer Networking (Weekend program)

Anomaly based intrusion detection system on semi-fog of Internet of Vehicle services by using deep learning approach

By: Berihun Desalegn

Advisor: Geletaw Sahle (PhD candidate)

Co-advisor:Mr.Temesgen (M.Sc.)

February 2023

Jimma, Ethiopia

Jimma University

Jimma Institute of Technology

Faculty of Computing and Informatics

Anomaly based intrusion detection system on semi-fog of Internet of Vehicle services by using deep learning approach

By: Berihun Desalegn

Approval sheet

This thesis work entitled Fog based anomaly detection model for Internet of Vehicle services using deep learning approach has been read and approved as meeting the requirement of faculty of computing in partial fulfillment for the award of degree of Masters of Science in computer networking, Jimma University, Jimma, Ethiopia.

Main advisor: Mr. GeletawSahle (PhD candidate)

Signature:______________ _______________________ Date: Jan 04, 2023

Co-advisor: Mr. Temesgen D

Signature:_____________________________________ Date: Jan 04, 2022_

Acknowledgment

First of all I would like to thank my principal advisor Mr.Geletaw Sahle and co-advisor Mr. Temesgen D for their assistance and they are open any time for providing the necessary comments. I thank Mr. kebebewAbabu and Mr.Getamesay Haile for their voluntary to give constructive ideas in the conducting of this paper and also others teachers under faculty of computing and informatics who teaches me a lot from starting to the end of this class. I would also like to show my gratitude to Mr.Kelemu Leykun for his support and giving correction on my paper and Mr. Daniel Alemu and Mr. TameneYilma for their financial support.

Abstract

The Internet of Vehicles (IoV) is a new paradigm of vehicular networks inspired by the adoption of the Internet of Things (IoT) in Vehicular Ad-hoc Networks (VANETs). There are mainly two types of applications used in IoV; safety, and non-safety applications. Because communication of IoV is vehicle-to-anything; it is compromised by different types of attacks, such as denial-of – service (DoS) attacks and impersonation attacks. DoS attacks create security problems in vehicles by flooding unnecessary messages and creating congestion that leads to the safety message being not delivered on time for the receiver vehicle.

Attackers can get the advantage of releasing fake or malicious data to the vehicular network by controlling either onboard unit (OBU) or roadside unit (RSU). Previously published paper deploys their deep learning classifier module either of OBU, edge server, or RSU. As they employ on OBU of vehicular network attackers can penetrate anomalous data by controlling RSU. When they deploy on an edge server there is a communication delay and need a high cost for adding an edge server in different locations. Lastly, they deploy on trusted vehicle network infrastructure RSU, but they don't use any alternative in the case of RSU stop working. By considering all the challenges of previous work, we are proposing anomaly based intrusion detection system on semi-fog of IoV by using deep learning algorithms. The deep learning algorithm is employed on RSU and OBU by selecting the vehicle as cluster head in case RSU stops working.

In this work, we are using two deep neural network models, MLP (multilayer perceptron) and LSTM (long short term memory), for training and testing a dataset. In this deep learning, we train and test our deep neural network model by using WSN-DS and by customizing it to our deep learning scenario. The dataset consists of an imbalanced dataset distribution and we are applying SMOTE (synthetic minority oversampling technique) to balance the dataset. We are training and testing the dataset after the SMOTE technique is applied to it to remove overfitting problems. This paper is a novel one in terms of applying the deep learning classifier module on the fog layer or RSU and OBU to a selected vehicle by selecting a cluster head in case RSU stops working. After applying the SMOTE technique, we got 99.1%, 98.8%, 99.3%, and 99.6% accuracies for flooding, TDMA, grayhole and blackhole attack respectively. For validating our deep learning model we use FC-BOT-IOT dataset and we got 99.70% accuracy.

Contents

List of figures

List of table

List of acronyms

AIDS	anomaly based intrusion detection system
CAM	co-operative awareness message
CAN	controller area network
CH	cluster head
DENM	decentralized environmental notification message
DL	deep learning
DNN	deep neural network
DoS	denial of service
DSRC	dedicated short-range communication
ECDSA	Elliptical curve digital signature algorithm
IDS	intrusion detection system
IEEE	institute of electrical and electronics engineers
IoT	internet of things
IoV	internet of vehicle
ITS	Intelligent transportation system
LSTM	long short term memory
MANET	Mobile ad hoc network
Mbps	mega bit per second
MLP	multi-layer perceptron

OBU	on board unit
RSU	road side unit
SMOTE	synthetic minority oversampling technique
VANET	vehicular ad-hoc network
VeReMi	vehicular reference misbehavior
VPKI	vehicular private key infrastructure
V2I	vehicle-to-infrastructure
V2P	vehicle-to-pedestrian
V2V	vehicle-to-vehicle
V2X	vehicle-to-anything
WAVE	wireless access in vehicle environment
WHO	World health organization
WSN-DS	wireless sensor network dataset

1 Chapter one

Introduction

The Internet of things (IoT) is an advanced technology that connects smart devices to the internet, such as the internet of vehicles, industrial robots, smart TVs, wearable's, wireless cameras, medical and healthcare devices and other electronic devices. The Internet of Vehicles (IoV) is a new paradigm of vehicular networks inspired by the adoption of the Internet of Things (IoT) in Vehicular Ad-hoc Networks (VANETs). Due to the rapid growth of vehicles on roads, transportation has a huge impact on economic and human assets.

It is projected that this increasing number of vehicles on the road will reach up to two billion or more in the coming decade[1] and as the number of vehicles becomes larger we will encounter an unfortunate rise in accidents, traffic jams, congestion, pollution and so forth. According to the road safety report released in 2021 by the World Health Organization (WHO), the road traffic death number globally has reached 1.3 million per year [2].

Intelligent transportation system (ITS) powered by vehicular ad hoc network (VANET) is one of the main goals for providing an improved driving experience and traffic efficiency for minimizing accident risk and traffic jams on the road by exchanging emergency messages and traffic forecasting respectively. VANET plays a vital role in autonomous cars by providing accurate information about road conditions, traffic security, resource scheduling. A new concept called the Internet of Vehicles (IoV) has recently emerged to overcome the limitations of VANETs (processing power to deal with large amounts of global information) and VANETs do not have the capability of analyzing, processing, and evaluating the global information that is acquired from the many vehicles that are a part of the network.

The IoV plays a vital role in communicating safety messages for safeguarding the driver, passengers, and the vehicle itself. In a high density road network that consists of a high number of vehicles on the road and each vehicle is recommended to broadcast a safety message to its one hope in the range of 100ms to 300ms [3], it is impossible to overcome the security problem in the internet vehicles by using the traditional way of verifying the safety message by using authentication and ECDSA algorithms. On the internet, vehicle intrusion or cyber security is very dangerous to human life unless they are detected or avoided by any mechanism before they disseminate the malicious information to the internet of vehicle network.

There are mainly two types of applications used in IoV; safety, and non-safety applications. Safety applications on the vehicular network are used to send safety messages. For example, various warning messages that assist vehicles on the road so that proper actions can be taken to prevent accidents and save people from hazardous situations. Safety messages include events such as road accident reports, traffic jam notifications, road construction reports, and emergency vehicle warnings [4].

The introduction of Fog Computing was as a means of extending that of cloud computing; thus, it inherited many of the privacy and security challenges faced by the cloud. Fog computing is an architecture that integrates cloud and IoT technology. The fog architecture acts like a cloud but is closer to the end user, and enables cloud computing facilities to be at the edge of the network, through which connected devices can obtain cloud services. On the internet of vehicle that has dynamic nature setting where the vehicle changes its network topology frequently and is unlimited to their mobility, it's difficult to secure their communication. Implementing deep learning on the fog layer of the internet of a vehicle is used to minimize the burden of checking whether the vehicle message is a malicious one or a valid one from a real cloud server that consumes high time and bandwidth.

Fog-based VANETs are capable of minimizing communication and computation delays [5], and stabilizing the networks by facilitating load sharing [6], load balancing and localized decision making. For exchanging information, a significant volume of data can be generated in the vehicular network and it is difficult all this data is processed and analyzed as malicious or normal data by using the resource installed onboard unit (OBU) of the vehicle.

Deep learning refers to a machine learning technique using an architecture comprising a number of hierarchical layers of non-linear processing stages. DL is a powerful tool used to analyze huge traffic volumes and accurately distinguish the normal and abnormal behavior of different systems by extracting deep complex patterns from raw data (packets)[7]. Vehicles are a limited resource for computation and storage of big data analytics that is forwarded or broadcast by many vehicles on the vehicular network and it's inappropriate for implementing this complex DL on board of vehicles. Implementing deep learning on fog layer can improve the analyzing and computing large volume of vehicular data on fog nodes with low latency and high response time.

DL can be implemented on the fog layer or road side unit (RSU) where fog computing allows the implementation and execution of attack detection in a distributed, powerful and scalable manner. Recently, DL has been used in cyber-attack detection because of its capability of extracting and learning deep features of known attacks and detecting unknown attacks without the need for manual feature engineering. Deep learning has been used for intrusion detection with the Internet of Things, especially in the Internet of Vehicles.

In this paper we are proposing anomaly based intrusion detection system on semi-fog of internet of vehicle by using deep learning algorithm. In this scenario, we are putting deep learning on the fog layer of the internet of the vehicle (road side unit) and selecting the OBU of vehicle for identifying the normal and abnormal behavior of the vehicle. Deep learning is deployed on RSU on vehicular network systems because any communication can be reached to the RSU and also deployed on selected vehicular node in the case of RSU fails and vehicles are out of range of working RSU.

1.1 Motivation

Millions of people lose their lives on the road each year due to accidents. According to a WHO survey published in 2021[2], around 1.3 million people die every year because of road accidents. The critical problem of this accident is due to getting the necessary safety message on time and there were security problems in the vehicle network and malicious data was disseminated. Recently, as the number of vehicles on the road increases, accidents and road congestion has become a big problem in the transportation system and this leads to the loss of a large number of human lives and properties.

To cope with the problem, an intelligent transportation system has been developed by academic and vehicular manufacturers to create communication between vehicles by using onboard unit (OBU) to improve road safety and driving comfort. Security issues are the main problem with the internet of vehicle as the vehicles that join the IoV increase and its significant challenge is to identify normal data from abnormal data on the vehicle network by using traditional security mechanisms.

Our work can fill out the problem of security in communication of internet of vehicle by applying Anomaly based intrusion detection system on semi-fog of Internet of Vehicle services by using deep learning approach.

This work identify or detect the data exchanged in the network is normal or abnormal before disseminated in the network and if the traffic is abnormal it block the traffic from sharing in the network and tell for all vehicular node not use any message from untrusted vehicle.

1.2 Statement of the problem

Communication in IoV is not limited to vehicle-to-vehicle and vehicle-to-other infrastructure like VANET. The communication is vehicle-to-anything and, as communicating parties increase, an attacker can get a hole to penetrate anomalous message in the network. In vehicular network the security risks increase with a rapid growth in the connectivity of smart vehicles [8]–[10]. In the scenario of IoV only vehicles are registered and other communicating parties like pedestrians, buildings, homes, grids and others can create communication without authentication.

Due to more communicating party is participated in IoV network attacker can control the OBU as well as RSU for releasing fake information in the network. The communication of IoV network is compromised by different types of attacks, such as a DoS attack and an impersonation attack. DoS attacks are the most attack that creates security problems on vehicular network by flooding unnecessary messages that creates congestion and it leads to the safety message being not delivered on time for the receiver vehicle. Also they can fully or partially drop forwarded message before they delivered to the receiver vehicle. Messages exchanged in vehicular network are time critical.

For overcoming the problem of cyber-attack on the internet of vehicles, different machine learning and deep learning are used to identify malicious data from normal data by using intrusion detection and prevention mechanisms. From the overall reviewed papers we have identified the following three deployment techniques with gaps:

A lot of papers deploy their heavy deep learning classifier on resource constraint OBU and mostly, the message that comes from RSU is trusted in the vehicular network. As they deploy the deep learning classifier on the OBU, attackers can get advantage of releasing fake information to the network by controlling RSU. Secondly, research work like [11]–[14] deploy their deep learning on RSU and they achieve the best result by using different machines and DL algorithms. But they don’t give any attention if the RSU fails and stops working. Malicious messages are exchanged in the range of failed RSU. Lastly, in [15], [16] the deep learning classifier is implemented on the edge server and any communication of the vehicle node that reaches to RSU is sent to the edge server for processed in deep learning algorithm and that is not good for time critical data. This creates communication delays and it is not a cost-effective one because we need some edge servers in different locations for applying the deep learning classifier.

By identifying all the gaps in previous works similar to our thesis topic, we are designing and implementing anomaly based intrusion detection system on semi-fog of Internet of Vehicle services by using deep learning approach. This work detects anomalous data in the vehicular network before they are disseminated in the network by deploying the deep learning on RSU and selected OBU in case of RSU fail or stop working.

This paper generally answers the following questions:

- How to employ the DL algorithm on the semi-fog layer of fog computing to detect anomalies in IoV?
- How to improve the performance of anomaly detection in IoV network by using a deep learning algorithm in the semi-fog layer?
- How to increase the detection rate and decrease the detection time of anomalies by using deep learning?
- How to train and test the dataset by using the best deep learning algorithm?

1.3 Objective

1.3.1 General objective

The general objective of this paper is proposing and developing Anomaly based intrusion detection model on semi-fog of Internet of Vehicle services by using deep learning approach

1.3.2 Specific objectives

To accomplish or achieve the general objective of this paper, the following specific objectives will be followed:

- To gain more understanding in deep learning based on anomaly detection in the area of IoV previously worked papers are reviewed.
- To propose anomaly based intrusion detection system on semi-fog of IoV.
- To improve the efficiency of intrusion detection for effectively classifying the dataset.
- To train the dataset using a deep learning algorithm.
- To Test and evaluate the model.

1.4 Methodology

1.4.1 Literature review

To acquire basic knowledge and understanding of the research area, literature review plays a significant role. This enables the researcher to gather enough information and understanding from other works in the area and it enables him to find possible holes or gaps and untapped features from other related works. To fill those gaps, proper review of literature will be conducted (such as articles, books, workshops, journals, conferences). In this literature review, to understand and get the latest view about anomaly detection, recently published papers are accessed and share their concerns of view.

1.4.2 Data source and training

We are using a WSN-DS public dataset that was released for research purposes and we made some modifications to it and customized it according to our deep learning.

Kaggle is used to train the dataset using the preferred two deep learning algorithms, namely multi-layer perceptron (MLP) and long short term memory (LSTM). Kaggle provides enough space for training machine learning with big data having GPU accelerator.

1.4.3 Testing and evaluation

The accuracy of this deep learning system is evaluated by using accuracy, precision, recall and other evaluation metrics.

1.5 Scope and limitation

The scope of this paper is limited to proposing and implementing Anomaly based intrusion detection model on semi-fog of Internet of Vehicle services by using deep learning approach and the deep learning is implemented on road side unit and selected vehicular node of a vehicular network. The deep learning algorithm identifies and classifies the malicious data from normal data based on a dataset that is trained on it. This work uses supervised machine learning and does not include an unsupervised machine learning approach. This work only detects four types of DoS attacks by using a deep learning algorithm.

1.6 Significance of the study

The Internet of vehicle provides many applications for enhancing the safety of drivers and travelers on ITS. Based on the purpose and service they give on the vehicular network, there are mainly two types of application in IoV. Safety applications on the vehicular network are used to send safety messages, for example, various warning messages that assist vehicles on the road so that proper actions can be taken to prevent accidents and save people from hazardous situations. Safety messages include events such as road accident reports, traffic jam notifications, road construction reports, and emergency vehicle warnings. These applications are focused on decreasing the probability of traffic accidents and loss of life. Information shared by this application must be secured and not be malicious to safeguard the driver from accident. Non-safety applications give comfort for travelers and make the journey more enjoyable by providing service such as infotainment, gaming, internet service etc.

On the internet of vehicle for enhancing security, detecting intrusion is the main thing that gets high attention for increasing safety for drivers and decreasing loss of human life due to malicious data is shared among neighboring vehicles on the vehicular network.

Hence, the contribution of this work is providing improved intrusion detection on the internet of vehicles on fog layer of vehicular network RSU and OBU of selected vehicle by using deep learning to classify malicious data from normal data in a vehicular network and detect the malicious data before it is disseminated and used in vehicular network

1.7 Organization of the thesis

The rest of the paper is organized as follows: chapter two presents a literature review about vehicular networks and their nature of communication for exchanging time-sensitive messages and types of machine learning and deep learning algorithm for securing message exchange in vehicular networks. Chapter three introduces related works that are carried out to improve the security of vehicular network communication. Chapter four presents a survey of datasets used in vehicular networks. Chapter five presents the detail of the proposed solution and its architecture details how the proposed solution overcomes the existing problem. In Chapter six, the proposed system is implemented by applying the proposed algorithm. After implementing the proposed solution, we are evaluating our work by using experimental analysis. Finally, chapter six presents the conclusion, contribution and future works.

2 Chapter two

Literature review

Smart cities are the development vision to integrate information communication with internet of things and provide better quality of life to its citizens with aim of rapidly and continuously urbanization of their environment [17]. With the objective of the IoT, many different physical objects are connected to create communication in smart cities and vehicular nodes are one object that connect to the IoT and customized to internet of vehicles (IoV).

The Internet of Things has changed from the conventional small scale Vehicular Ad-hoc Network (VANET) to a highly scalable, manageable internet-based “Internet of Vehicles” (IoV). IoV is a vehicular network model consisting of vehicles, users and other smart devices connected to a network and aims to provide various safety as well as entertainment services. Vehicles on IoV system are equipped with different sensors that collect different types of data and send it to a computation unit for computation and analysis, based on which directions and other information is sent to each vehicle. The Internet of Vehicles (IoV) is a typical application of the Internet of things in the field of transportation, which aims at achieving an integrated intelligent transportation system to enhance traffic, to avoid accidents, to ensure road safety, and to improve driving experiences.

The Internet of vehicle is a new paradigm that integrates the internet of things with a vehicular ad hoc network and creates a large vehicular network where vehicles are communicating and gathering information about their environment for increasing safety for drivers and decrease the traffic accident for minimizing the percent of death [18].

The deployment of the IoV in smart cities enables information sharing and the gathering of big data information on vehicles, roads, infrastructure, buildings, and their surroundings. The IoV ecosystem can provide services for intelligent transportation applications to guide and supervise vehicles, and provide abundant multimedia and mobile Internet application services [19]. In internet of vehicle privacy and security gets a lot of attention for providing safety for drivers by applying security mechanisms in information exchange in the vehicular network.

Because their nature is complex in terms of high speed, they change their position frequently, design of topology and vehicular node communicate wirelessly it is difficult to secure vehicular network and it’s vulnerable to different cyber security attacks. Depending on the different nature of complexity of the internet of vehicles, many internal and external attacks are faced in vehicular network communication and because of those cyber security attacks many human and economic assets are loosed. The Intelligent Transportation Systems (ITS) main aim is to provide a solution for protecting passengers from possible accidents and traffic congestion problems.

2.1 Communication architecture of IoV network

There are two communication architectural models in vehicular network for providing enhanced transportation service to the drivers and passengers.

2.1.1 Intra vehicle communication

This communication model performs communications within the vehicle itself. Currently, modern vehicles have a group of sensors which are responsible for undertaking various tasks, such as checking inter-vehicle distance and road conditions, smoke and fire detection, vehicle acceleration/deceleration system, obstacle detection radar, and so on. There are different intra-vehicle communication systems:

- Vehicle to sensor communication (V2S): It is a communication type that provides communication between sensors in an intra-vehicle sub-network [20], [21].
- Device-to-device communication: in this type of communication, two or more devices directly connect and communicate with one another, rather than through an intermediary application server. These devices communicate over many types of networks, including IP networks or the internet. These devices use protocols like Bluetooth, Z-Wave, or ZigBee to establish direct D2D communications [22]–[24].
- Vehicle-to-driver communication: [25]

2.1.2 Inter vehicle communication

Provide communication between vehicle and vehicle as well as vehicles to their environments. There is different communication models included in this communication architecture:

- Vehicle-to-vehicle communication (V2V): Its communications architecture provides interaction within vehicles. In V2V, a vehicle can broadcast and exchange traffic conditions with other vehicles [22], [26]–[30].

- Vehicle-to-infrastructure communication (V2I): the communication type in which the information will be broadcast between the nodes (i.e. vehicle) and the infrastructure (known as ITS), to deal with important information such as road conditions and safety events that have been taken into account. In this V2I, a vehicle (node) launches a connection between RSU and contact with external networks, which is the internet [31]–[37].
- Vehicle-to-everything(V2X):Vehicle-to-everything(V2X) communications include vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), vehicle-to-pedestrian (V2P) , Vehicle-to-home(V2H), Vehicle-to-barrier(V2B) and vehicle-to-network (V2N) communications, and support a variety of applications such as forward collision warning, lane change warning, multimedia content sharing and other traffic safety-related applications[38]–[41].
- Vehicle-to-pedestrian (V2P) - is a type of communications that supports awareness for vulnerable road users like pedestrians and cyclists and its communication type in which vehicles share and communicate important information with passengers[42]–[44].
- Vehicle-to-roadside unit (V2R): is a communication type in which vehicles communicate to the road side unit to provide an efficient service to the user and create a large vehicular network [21], [45]–[48].
- Vehicle-to-home (V2H) -The concept of vehicle-to-home (or V2H) is similar to vehicle-to-grid (V2G). The difference is, however, that with V2H, the energy is used to power a home instead of being delivered back to the grid. Vehicle-to-home gives homeowners control over their energy usage by allowing them to shift some of the power they use from the grid to off-peak times when energy prices are lower[49].
- Vehicle-to-barrier (V2B): is communication between vehicles and radio embedded road side barriers and the communications established between vehicles and radios embedded in roadside barriers will ensure that vehicles are kept on the road as well as to mitigate ROR crashes[50].
- Vehicle-to-Grid (V2G): V2G systems bring many benefits to power systems, such as stabilizing energy demand and supply fluctuations and assisting Plug-in Electric Vehicle (PEV) users in reducing energy costs[50]–[54].

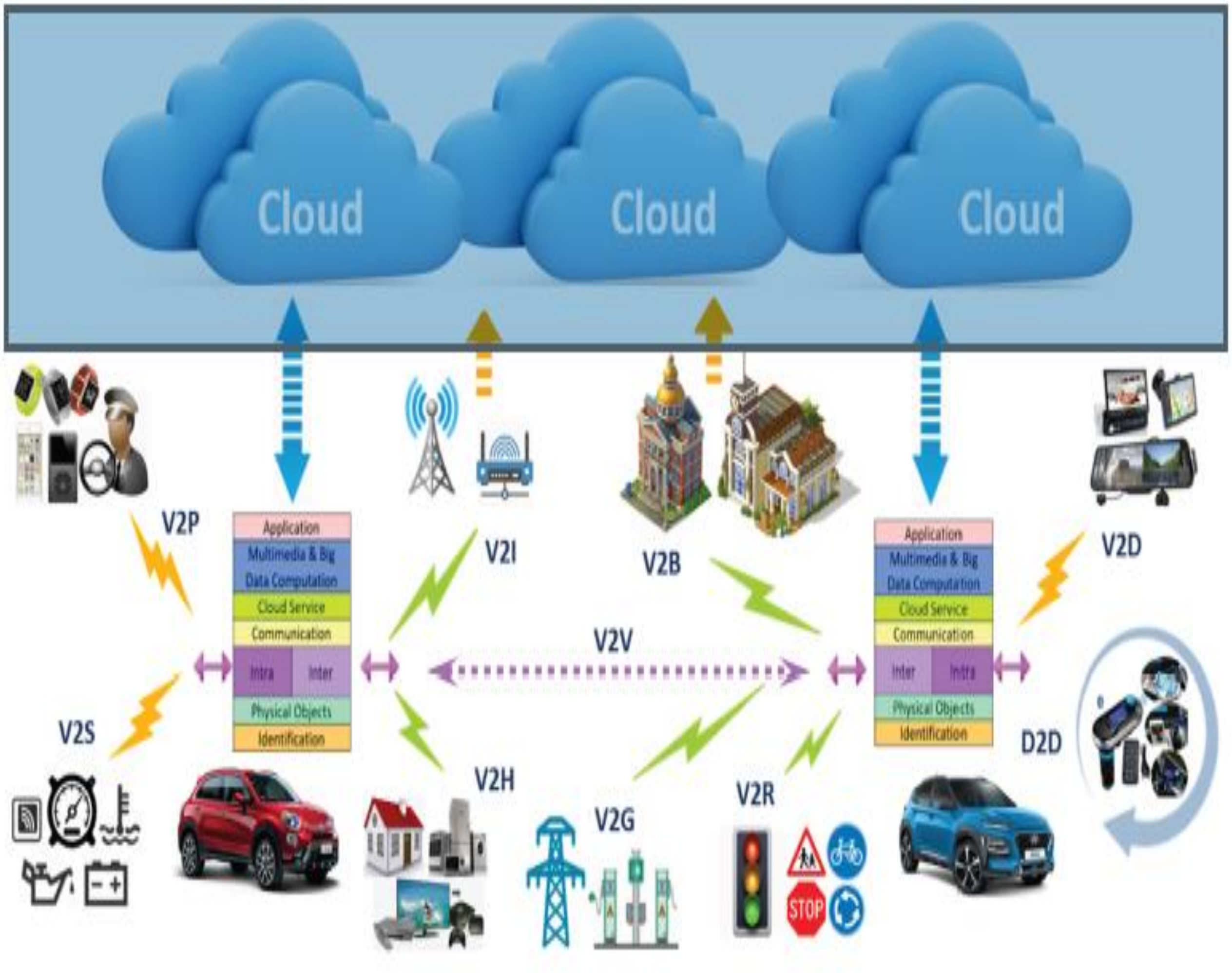

Figure 1: universal IOV communication[19]

2.2 Characteristics and challenges of IoV

2.2.1 Characteristics

Vehicular network is composed of different vehicle nodes that differ from other wireless nodes and due to those various characteristics it’s difficult to design of internet of vehicles technologies. Some of the characteristics will bring challenges to IoV technologies. VANET is a sub-part of MANET application which has its own distinct characteristics [55] explained as:

High mobility: the nodes in a vehicular network can move at high speed and their mobility is not limited like MANET [56], [57].

Dynamic topology: The node in IoV is highly mobile and the speed of the vehicle should also be random so that the node position will change frequently. The topology is changed rapidly and unpredictable. It facilitates the entire network becoming prone to attacks and makes it hard to find misbehavior in the vehicular network [57].

Time-critical: Within time, the information in IoV should be sent to the accurate node, so that the node will make a decision and execute action correspondingly [57].

Frequent exchange of information: Normally, the VANET is ad-hoc in nature. It inspires the nodes to collect information from neighboring vehicles and roadside units. So, the nodes exchange their information periodically [56]–[58].

High scalability and heterogeneity: in a big city or wide range vehicular network there is millions of different heterogeneous vehicular nodes can join the ITS. This leads that the IoV requires a large scale network and this network must be highly scalable for accommodating the newly joined vehicular nodes or the increasing number of vehicles to vehicular network.

Limited bandwidth: In VANET, the standard DSRC band should be measured as limited. The bandwidth of the DSRC band was 27 MHZ. The throughput was 27 Mbps, which is a theoretical value [57]

Safety critical application: in vehicular networks, the communication message holds sensitive data that protects the driver from accident [56], [57], [59].

2.2.2 Challenges in IoV

In a vehicular network we are facing some challenges in preserving the communication of each individual node in the network [60], [61] and discussed as follows:

Security and privacy: Keeping a reasonable balance between security and privacy is one of the main challenges in IoV[62]–[64]. The receipt of trustworthy information from its source is important for the receiver in the vehicular network.

Poor network connectivity and stability: Due to high mobility and rapid changes of topology, which lead to frequent network disconnections and link failures, message loss should be common. Then, how to elongate the life of communication links is always challenging.

Hard delay constraints: Many IoV applications have hard delay constraints, although they may not require a high data rate or bandwidth. For example, in an automatic highway system, when a brake event happens, the message should be transferred and arrive at a certain time to avoid a car crash. In this kind of application, instead of an average delay, a minimal delay would be crucial.

High reliability requirements: transportation and driving-related applications are usually safety-sensitive. Obviously, such an application requirement is high reliability. However, due to complex network architecture, large network scale, and poor stability of network topology, achieving high reliability in IoV is far from trivial. A special design should be conducted in various layers, from networking protocols to applications.

High scalability requirements: High scalability is another big challenge in IoV. As mentioned before, IoV is usually very large in terms of node number and deployment space. Such a large scale certainly requires high scalability in IoV technology. .

Service sustainability: Assuring the sustainability of service provided in IoV is still a challenging task, calling for high intelligence methods, as well as a user-friendly network-mechanism design. There are challenges in adjusting all vehicles to provide sustainable services over heterogeneous.

Applications Heterogeneity: IoV has various applications of safety and non-safety applications. These safety applications are time-sensitive that need low latency and high reliability, while non-safety applications need low packet loss, better throughput and higher utilization of the resource. Therefore, designing an efficient and effective communication technique that can satisfy the demands of applications requirements is a critical issue in IoV.

2.3 Application of IoV

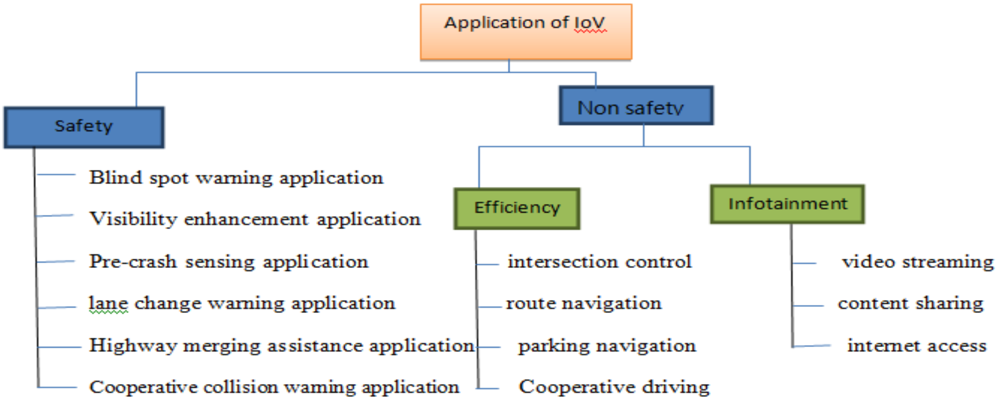
Application of IoV
Safety
Non safety
Efficiency
Infotainment
Blind spot warning application
Visibility enhancement application
Pre-crash sensing application
lane change warning application
Highway merging assistance application
Cooperative collision warning application
intersection control
route navigation
parking navigation
Cooperative driving
video streaming
content sharing
internet access

Figure 2: Application of IoV

2.3.1 Safety application

Safety applications are applications that have the main focus to decrease the probability of traffic accidents and loss of life[65]–[67]. A significant number of accidents happening in the world every year related to intersections, blind-spot, rear-end and lane change collisions. Safety applications use information collected by vehicles from its neighboring vehicles to alert a driver to prevent such collisions with other vehicles. Some examples of safety applications are as follows:

- Blind spot warning application: This application is designed to alert a driver when there is a vehicle at the blind spot when a vehicle wants to change lanes.
- Visibility enhancement application: This application is used for alerting a driver when there is an unsafe situation occurring when there is low visibility due to heavy rain, fog, storm, or others.
- Cooperative forward-collision warning application: It is an application which designed to alert a driver when there is a potential of rear-end collision to vehicle ahead. In general, the application uses position, velocity, acceleration, heading, and yaw-rate to analyze the unsafe situation.
- Lane change warning application: This application is used to alert a driver when there is a potential collision for changing lanes. When a driver wants to change lanes and uses a signal for changing lanes, the vehicle uses the information of other vehicles such as position, velocity, acceleration, and heading to analyze the situation such as calculating the gap between vehicles for safe lane changing.
- Highway merging assistance application: Alerts a driver when a vehicle at the blind spot or a vehicle is on a highway ramp trying to merge. The vehicle uses the heading, position speed of that vehicle to analyze the situation and alert a driver if there is an unsafe situation.
- Cooperative collision warning application: It alerts a driver when there is a potential accident about to happen. The application uses the collected information of neighboring vehicles such as position, speed, acceleration, wheel angle to analyze them with its sensor information for a potential collision.
- Pre-crash sensing application: Far way vehicle becomes active when there is an accident about to happen with a vehicle. This application uses neighboring vehicle information to detect this kind of situation.

There are also other types of safety applications in IoV with a goal of improving road safety and to reduce the number of road accident. Like emergency call, auto breaking, speed control, car surveillance, driving behavior analysis, car self-diagnosis, real time traffic information etc…

2.3.2 Efficiency application

Efficiency is one of the major concerns of transportation management. Vehicular network technology brings new possibilities of efficiency improvement. The following are some of efficiency related application in internet of vehicle:

Intersection control

Traffic control at intersections has been always a key issue for ITS. The key point is how to schedule traffic signals efficiently, according to traffic volume information, so as to reduce waiting time and improve fairness. Most existing work on intersection control is traffic-light based, and the key issue is to determine a good signal-scheduling plan.V2I-based traffic-light scheduling is widely studied in [68]–[71].

Route navigation: Vehicular network-based navigation is studied to avoid the drawbacks of GPS-based or similar navigations[72]–[74].

Parking navigation: Finding an available parking space in an urban environment with the help of vehicular networks is also an interesting problem[75], [76].

Comfort-based IoV applications aim to give additional information to drivers to make the trip comfortable and enjoyable. This may include weather, information on parking lots[76].

2.3.3 Infotainment service

Infotainment services include mainly Internet access services and file sharing among vehicles, especially video sharing. Vehicle-to-Internet communication is a challenging task. A QoS framework to ensure data forwarding to the Internet in a gateway-free area in a highway scenario is proposed in[77]. It consists of a proxy-based Vehicle-to-Internet protocol, with a prediction-based routing algorithm and IEEE 802.l1p EDCA scheme. Video streaming over VANET has attracted more and more attention. Asefi et al.[78]introduced a quality-driven scheme for seamless delivery of video packets in urban VANET scenarios, which includes routing and mobility-management mechanisms based on Mobile IPv6.

2.4 Types of message used for safety application in IoV

SAE J2735 over Dedicated Short Range Communication (DSRC) is a standard for messaging in the Vehicular network. In European standard there are two types of safety message in vehicular communication [79]. Thus, in general, safety messages can be categorized into two groups. These are co-operative awareness message (messages that are transmitted for awareness of the environment) and the other messages are decentralized environmental notification message (event messages which are triggered by unsafe situations).

2.4.1 Co-operative awareness message (CAM)

Which is periodically broadcasted by vehicles to indicate their speed and position and it also provide a basic awareness service in cooperative ITS networks, by means of periodic sending of status data to neighboring nodes? This service can be viewed as an application support facility in charge of periodically distributing messages containing information of presence and location, as well as basic status. CAM messages are disseminated to neighboring ITS stations that are located within a single hop distance from the sender. By receiving CAM messages, the ITS station is aware of other stations in its neighborhood area as well as their positions, movement and relevant characteristics. The receiver of a CAM message is expected to evaluate the relevance of the information contained within the message. This allows ITS stations to get information about its surroundings and act accordingly.

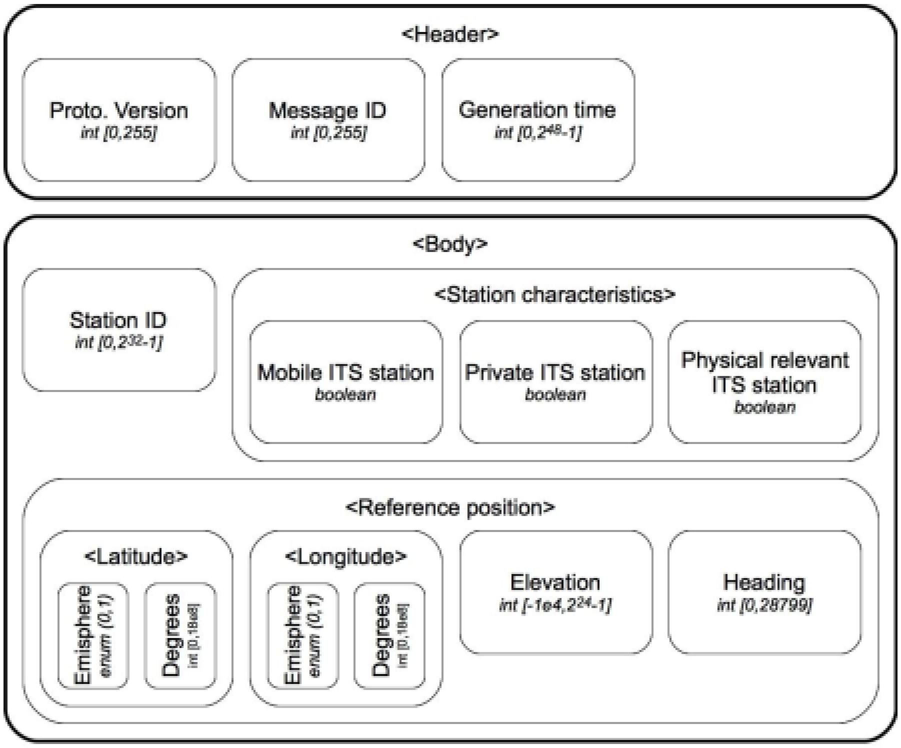

Figure 3: Structure of CAM message[79]

2.4.2 Decentralized environmental notification message (DENM)

That is send when hazardous event occurs and forwarded to warn vehicles over a wide area. A DENM transmission is triggered by a certain ITS application that detects a relevant driving environment or traffic event. The application solicits to the DENM messaging facility the transmission of DENM messages notifying about the event. According to the standard specification, an event is characterized by an event type, which is an identifier associated to the type of event detected (e.g. vehicle breakdown, traffic jam, etc.); an event position describing either a concrete position or geographical area; the event detection time, which is an expiration time representing when the event is expected to be terminated; a destination area indicating the geographical area over which the DENM message needs to be disseminated among ITS stations; and a transmission frequency of the DENM messages issued by the same ITS station.

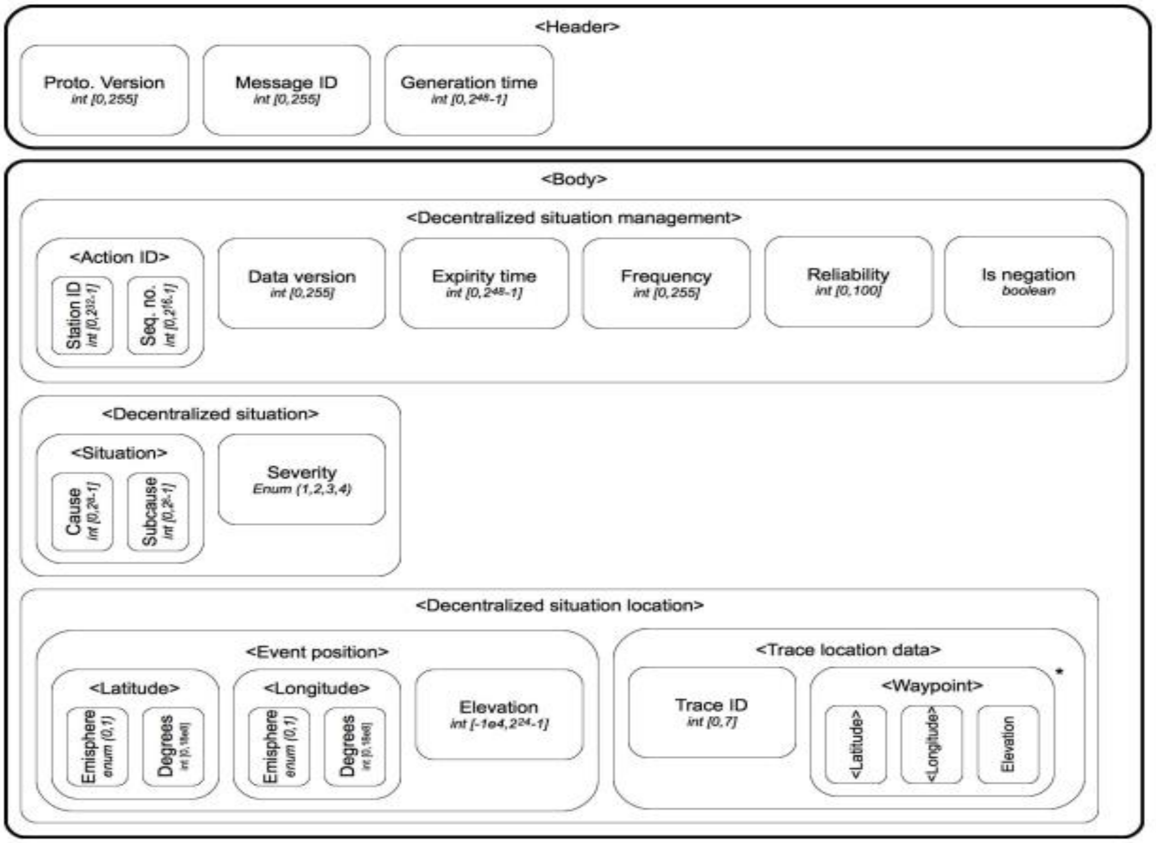

Figure 4: Structure of DENM message[79]

2.5 Internet of vehicle security and cyber attack

Internet of vehicle is one of the sub categories of IoT in which vehicular nodes are connected and communicate each other using wireless communication. Wireless communication is more susceptible to different attacks than wired communication. Though vehicular communication is fully uses wireless medium to exchange information between node and depending on the way of information exchange between vehicles in the network it needs high security mechanisms for protecting the information exchange in the vehicular network from different attacks and to identify misbehaving vehicles in the network.

Implementing security techniques in internet of vehicle is critical one for safeguarding the life of the driver and passengers by timely delivering secure message for mitigating accidents on the road and minimizes loss of human life and properties by traffic accident. In vehicular network vehicular nodes communicate through wireless medium and this opens a door for serious attack in the network such as Sybil attack, DoS attack etc. A Sybil attacker can create and manage multiple phony identities which share false information in the network in order to craft a false impression of nonexistent events.

Safety applications are one of the most important and promising advantages of IoTs. However, they are time critical applications and require data transmissions from one vehicle to another vehicle at the right time. In timing attacks[80], when malicious vehicles receive a message, they do not forward it as normal but add some timeslots to the original message to create delay. Thus, neighboring vehicles of the attackers receive the message after they actually require or after the moment when they should receive that message or they get the message after some event occurs.

A **cyber-attack** is any type of offensive action that targets computer information systems, infrastructures, computer networks or personal computer devices, using various methods to steal, alter or destroy data or information systems. A security attack can be defined as any type of illegal procedure that focuses on computer information systems, infrastructures, computer networks, or personal computer devices. An attacker is a person or process that tries to access data, functions, or other restricted places of the system without authorization, potentially with malicious purpose.

A cyber attacker may steal, alter, or damage a given target by hacking into a susceptible system. Security attacks can range from installing spyware on a personal computer to trying to destroy the infrastructure of entire countries. Legal experts are asking to limit the use of the term to incidents causing physical damage, recognize it from the more routine data breaches and wider hacking activities.

2.5.1 Types of attack in IoV

According to[81] there are different types of attacks are facing vehicular network communication.

- ➢ Insider vs. Outsider If the attacker is a member node who can communicate with other members of the network; it will be known as an Insider and able to attack in various ways. Whereas, an outsider, who is not authenticated to directly communicate with other members of the network, have a limited capacity to perform an attack (i.e., have less variety of attacks).

- Malicious vs. Rational: A malicious attacker uses various methods to damage the member nodes and the network without looking for its personal benefit. On the contrary, a rational attacker expects its own benefit from the attacks. Thus, these attacks are more predictable and follow some patterns.
- Active vs. Passive: An active attacker can generate new packets to damage the network whereas a passive attacker only eavesdrop the wireless channel but cannot generate new packets (i.e., less harmful).
- Local vs. Extended: An attacker is considered as local if it is limited in scope, even if it possesses several entities (e.g., vehicles or base stations). Otherwise, an extended attacker broadens its scope by controlling several entities that are scattered across the network.

2.6 Security counter measure in IoV

Physical detection: The main idea behind this defense is to put radar or signal receiver that detects the physical existence of the vehicles around it. Then, after performing some calculations on the message and fulfilling certain criteria, the message is accepted or rejected[82], [83].

Intrusion detection system (IDS) is an important supplementary measure of network security. IDSs provide protections against internal and external attacks by collecting and analyzing information from internal network systems to check if there exist system behaviors which violate security strategy or signs of attack. Signature-based detection and anomaly-based detection are the two main classes of detection methods[84].

Signature-based malware detection: By analyzing the malware, a signature can be produced which can be used later to detect that malware[85].

Reply protocol: When the vehicle receives a message, the receiver will send it to the RSU to check the correctness of the message, and that the sender is not malicious.

Trust models: Create a trust model that will evaluate the truthfulness of the message and the vehicle that sent the message and according to the level of trust, it can be established whether to accept or reject and discard the message[86], [87].

Event-based reputation system (EBRS) can defense against multi-source Sybil attacks, to ensure the integrity and preserve the privacy of vehicles. By establishing a reputation and trust threshold for each vehicle message, then the false message is restricted to legitimate identities. In EBRS, a trusted RSU is used to as CA[88].

Key management

Elliptical curve digital signature algorithm (ECDSA): This algorithm uses a digital signature along with hashing and public key to provide authenticity in the system. Both the sender and the receiver need to agree on elliptical curve domain parameters[89], [90].

Cryptographic digital certificates: One such example is vehicular public key infrastructure (VPKI), in which before sending a message, the vehicle must cryptographically sign it with its private key and the receiver will decrypt it with the sender's public key that it can get from certification authority (CA). In this way, the receiver is able to authenticate the message as well the sender[89], [91]–[93]

VPKI (vehicular private key infrastructure): Relying on the public key encryption method, each car will have its own public and private key along with CA in order to authenticate the cars and the message

2.7 Intrusion detection system

IDS are either a computer program or hardware that mechanizes intrusion detection, monitors network activity for suspicious activities, and sends notices to an administrator. An IDS can be a piece of installed software or a physical appliance that screens network activity in order to identify undesirable action and events such as illicit and pernicious activity, activity that abuses security policy, and traffic that abuses acceptable utilize policies and this intrusion detection system (IDS) is program that mechanizes the intrusion detection process.

Cyber security for any network can be divided into internal and external attackers. External attackers can be easier to prevent through cryptographic means, but internal attackers are assumed to have access to authentication for the system. For IoVs, internal attackers are assumed to have access to an enrollment certificate from SCMS as well as the pseudonym certificates that go with it. In order to protect against internal attackers, it is necessary for a VANET to prevent attacks as well as be able to identify and respond to attackers.

Classification of IDS

2.7.1 Based on location of deployment

There are three different types of intrusion detection system depending on location of deployment of IDS.

- **Network based intrusion detection system (NIDS)** act at high level of the network system and it will capture network traffic and analyzes the passing activity for assault before sending it to the end users. Once anomaly behavior is detected it can send alarm to administrator. NIDS can distinguish 4 major sorts of assaults: denial of services, Test, user to root and remote to user.
- **Host based intrusion detection system (HIDS)** typically act at the host level and will evaluate things like log files from the OS, and services and software executing on the host. HIDS is single computer particular intrusion detection framework which screens the security of that framework or computer from inner and outside assaults. The inner assaults allude to the circumstance where it recognizes which program gets to which asset and is there any security break.
- **Hybrid intrusion detection system-**combines the network based IDS and host based IDS together and detect anomalies.

2.7.2 Based on decision making approach

Intrusion detection system can be centralized or distributed/decentralized based on approach of making decision

- **Centralized IDS-** have multiple agents analyze an issue and report to central command and control (C&C) which will then decide upon a course of action.
- **Distributed IDS-**it can be distributed to analyze issues in different geo location and they can decide the incoming data is misbehave or not and send it to central IDS for further processing. Distributed IDSs will either be a Collaborative Distributed IDS (CDIDS) which is a system of agents that work together to collect data and make decisions collectively, or a single standalone IDS.

2.7.3 Based on detection mechanism

Intrusion detection system can be divided in to three based on detection mechanisms.

- **Anomaly based**

A model will seek to learn a profile or heuristic for normal behavior and then detect when there is a deviation from normal behavior. This has the benefit of being able to detect zero-day attacks since they will be outside the normal behavior of the system but will also have a number of false positive results. Zero-day attacks are cyber-attacks who are not currently known and are being launched at a system for the first time ever. Anomaly based intrusion detection framework is based on the network behavior. The organize behavior is characterized by the administrator or is learned by the dataset during the preparing stage of the advancement of IDS. Rules are characterized for ordinary behavior and anomalous behavior.

- **Knowledge based**

Knowledge-Based IDSs, on the other hand, match attack patterns and signatures to those already found in a database. Because of this, they are able to have very low false positive rates, but also are unable to detect zero-day attacks. In signature based detection mechanism the attack designs are spared within the database and each parcel of the organize activity is compared with the assault designs to distinguish unusual behavior. Of the Knowledge-Based Techniques employed in VANETs, Rule-Based reasoning and Case- Based reasoning are the most common. Rule-Based reasoning is uses a series of if-then logic chains which are set by humans manually. This is often the same as writing code which will check for certain requirements. Case-Based reasoning will choose the best outcome for a situation based upon similarities to past situations. However, this requires a history log to be kept of past situations which may not be possible for the resource-constrained OBUs.

- **Hybrid detection** technique will combines the above detection techniques for increasing accuracy of anomaly detection and identifying attacks easily and decrease false alarm rate.

2.8 Deep learning methods and algorithms

Deep learning, a subgroup of machine learning, focuses on simulating the way a human brain works to learn from experience (i.e., large volume of data) by employing neural networks with multiple layers[94]. Deep learning involves the use of complex models that exceed the capabilities of machine learning tools such as logistic regression and support vector machines, but this deep learning models are essentially “function approximators”.

Deep learning uses supervised learning in situations such as image classification or object detection, as the network is used to predict a label or a number (the input and output are both known) and it labels the input dataset for reducing error rate.

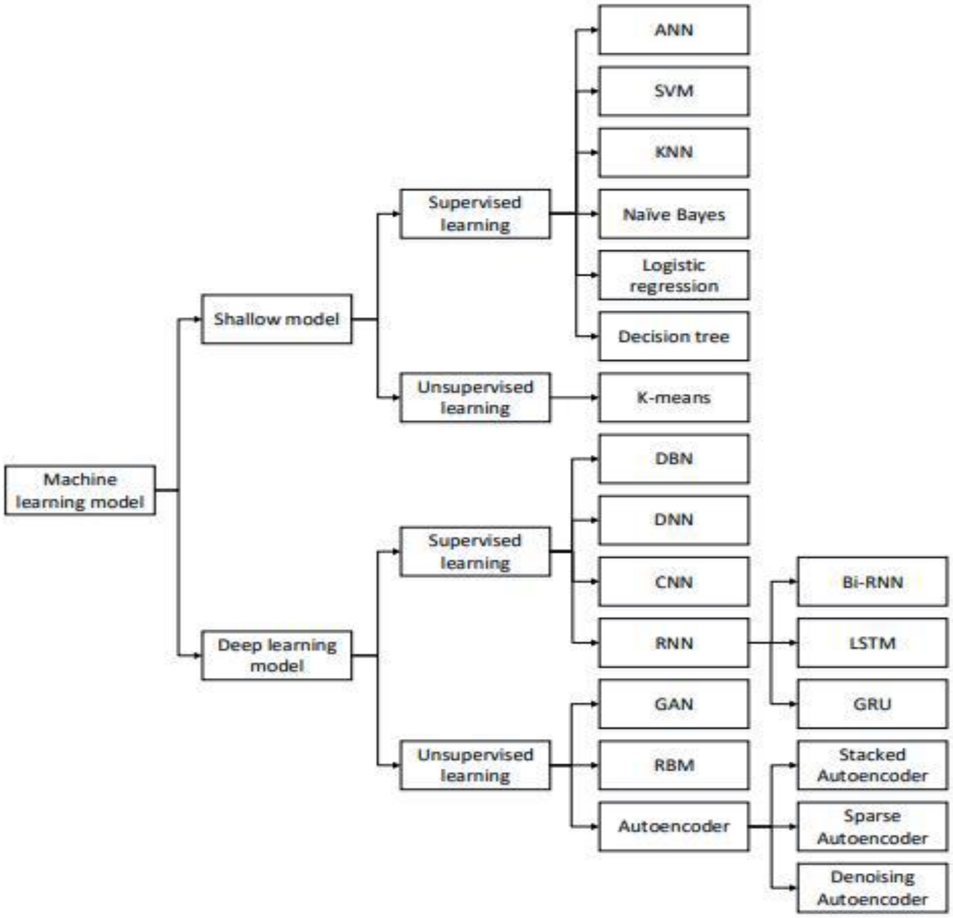

Figure 5: Taxonomy of machine learning and DL algorithm[95]

In our paper we are selecting two deep learning algorithm namely multilayer perceptron (MLP) and long short term memory (LSTM) for detecting anomalous communication in IoV.

3 Chapter three

Related works

As we discussed in chapter two, there are different employment methods of deep learning for detecting anomalies or intrusions in vehicular networks.

Some papers deploy their deep learning on resource-constrained vehicles for detecting the dissemination of anomalous messages in a vehicular network. Rasika S et al. Proposed an intrusion detection system for vehicular ad hoc networks using deep learning and they use a deep belief network (DBN) algorithm for detecting anomalies in the vehicular network [96]. In their approach, the output layer consists of two types: binary classification and multi-class classification, where the binary classification gives outputs whether there is an attack or not, and in the multi-class classification they give the attack name if an intrusion is detected in the network. They are using the CICIDS 2017 dataset that consists of benign and up-to-date common attacks for training and testing their deep learning algorithm and they get 98.07% detection accuracy.

Di Ma and Linxi Zhang proposed a hybrid approach toward efficient and accurate intrusion detection for in-vehicle networks and they are applying the advantage of rule-based intrusion detection and machine learning-based intrusion detection approach in their implementation [97]. They are combining the advantages of the two detection approaches to increase the accuracy and efficiency of anomaly detection. In the first stage, they are using rule-based intrusion detection which is used to quickly catch attack messages that violate the established rules, and in the second stage, they deploy the DNN based detection which is used to detect attack messages that fall outside of the scope covered by the rules in the first stage.

The first stage can reduce the workload of the second stage and further improve the efficiency of the hybrid IDS. In their approach they are collecting data by using the CAN Analyzer tool from the on-board diagnostic standard II (OBD-II) interface and for showing their proposed IDS is not limited to the specific vehicle model and for checking its applicability they are experimenting on four datasets collected from different vehicle models of different manufacturers (Honda Accord 2006, Honda Civic 2018, Ford Fusion 2013 and Chevrolet Volt 2013). Agrawal et al proposed and design a novel anomaly detection system for Intra-Vehicular Networks for detecting anomalies by using CNNs and LSTMs on the CAN network traffic of the vehicle[98]. The proposed framework consists of two components: a time series model which is a trainable deep learning model that learns features such as message frequency, data length, etc. from normal sequences for reconstruction; followed by an anomaly detection module that analyses the reconstruction error to classify a sequence as anomalous or non-anomalous. The underlying assumption is that an anomalous sequence will show deviation from the standard behavior and hence, produce a larger reconstruction error than a non-anomalous sequence.

The workflow consists of two phases: a training phase, where sequence features are learned and a threshold on the reconstruction error is determined, followed by a testing phase. The underlying deep learning models are trained on normal CAN data, and using a reconstruction thresholding approach the incoming CAN data is classified as genuine or anomalous. The detection system is trained on a cloud server and is to be deployed on the CAN bus system. The system first collects a sequence of messages and then passes it through the anomaly detection model, which consists of a re-constructor followed by a detection module. The proposed framework was evaluated against the CAN car hacking dataset. The main drawback of those articles is they employ the heavy deep learning classifier module on resource constraint OBU of a vehicle and attackers can release fake messages to the network by controlling the trusted network device RSU.

To solve the above challenges of detecting anomalous communication in vehicular networks by employing the deep learning classifier module on resource constraints of OBU, some scholars propose and design anomaly detection on the edge or fog server of a vehicular network. Harsh Grover et al propose edge computing and deep learning-enabled secure multi-tier network for the internet of a vehicle that secures the communication of the internet of vehicle based on deep learning classification on edge server [15]. They use two deep learning architecture approaches and they integrate stacked LSTMs that are created using multiple layers of LSTMs stacked one after the other and the CNN-LSTM algorithm that is intelligent combination of CNN and LSTM layers. They propose two different deep learning-based classification approaches for intrusion detection: Coarse-Grained Classification Method: This approach is a two-class coarse-grained classification method (CGCM) that distinguishes normal vehicle data from misbehavior data. In this approach, the faults and attacks are grouped into a single misbehavior (Faults + Attacks) class.

When the input data is passed through the CGCM classifier, every possible fault and attack type data is classified into the misbehavior class, while the normal vehicle data is classified into the normal class. Fine-Grained Classification Method: The second approach is a fine-grained classification method (FGCM) with three predicted classes based on normal vehicle behavior, faulty behavior, or attack behavior. Thus, compared to the first approach, this is a more fine-grained classification approach. When the input data is passed through the FGCM classifier, normal vehicle data is classified into the normal class, the fault type data into the fault class, and the attack type data into the attack class. In this approach, they train and test by using the VeReMi extension dataset, which is a more convenient dataset recently used in a vehicular network.

Alladi et al proposed Securing the internet of vehicles: A deep learning based classification framework [16]. In their implementation, they use CNN and LSTM algorithms on edge servers for the detection of anomalous data distribution in a vehicular network. The challenge of employing deep learning on the edge or fog server of cloud computing is it creates communication delays in time-sensitive message exchange and needs extra edge servers in different locations for deployment of the deep learning algorithm.

Lastly, by observing the above paper challenges, some scholars propose and implement their deep learning classifier module on trusted RSU.

Alladi et al proposed and implement a deep neural network for securing IoT enabled vehicular Ad-hoc networks [11]. They deploy their deep learning classifier module on the RSU of a vehicular network. In their implementation, they apply the CNN-LSTM algorithm and train and test their deep learning by using the VeReMi dataset extension.

On the other hand, Nie et al proposed data-driven intrusion detection for intelligent internet of vehicles: a deep convolution neural network-based method [12]. And Yurkovic proposed RSU-based intrusion detection and autonomous intersection response systems [13]. The work of the two papers is employed on RSU of vehicular networks for detecting anomalous data exchanged in a vehicular network. When they employ their deep learning on RSU of a vehicular network, they don’t put in alternative solution when RSU fail to work. If an RSU fails, in the domain of failed RSU abnormal data can be exchanged.

Approach	Employment	Dataset	Challenges or gaps
DBN[96]	OBU	CICIDS 2017	They implement the heavy deep learning classifier on resource constraint OBU of vehicle and attacker gets advantage of releasing fake information to the network by using the trusted RSU
Hybrid (Rule based and DNN)[96]	OBU	Their own dataset	They implement the heavy deep learning classifier on resource constraint OBU of vehicle and attacker gets advantage of releasing fake information to the network by using the trusted RSU
CNN and LSTM[98]	OBU	Car hacking	They implement the heavy deep learning classifier on resource constraint OBU of vehicle and attacker gets advantage of releasing fake information to the network by using the trusted RSU
Stacked LSTM and CNN[15]	Edge server	VeReMi	Create communication delay in time critical information exchange and extra cost for buying edge servers
CNN and LSTM[16]	Edge server	VeReMi extension	Create communication delay in time critical information exchange and extra cost for buying edge servers
CNN[11]	RSU	Their own dataset	They doesn’t give alternative solution in case of RSU fail to work and at that time in the failed RSU range malicious data is exchanged in the network
SVM,K-nearest neighbors,decision tree and	RSU	VeReMi	They doesn’t give alternative solution in case of RSU fail to work and at that time in the failed RSU range malicious data is exchanged in the network

random forest[13]			
CNN-LSTM[12]	RSU	VeReMi extension	They doesn’t give alternative solution in case of RSU fail to work and at that time in the failed RSU range malicious data is exchanged in the network
This work (MLP and LSTM)	RSU and selected OBU	WSN-DS	

Table 1: Summary of related work

3.1 Summary

To our knowledge of securing communication on the vehicular network, we are raising some gaps that need work to overcome the problem of exchanging malicious data on the vehicular network and creating accidents. The best way of decreasing accidents and loss of life is securing the communication of the vehicular network from attackers by using machine learning and deep learning mechanisms for the prevention and detection of intrusion or anomalies. In this paper, two different deep neural network algorithms are integrated to achieve the security of data exchange on the internet of vehicles and to identify normal data from malicious ones in the network. In vehicular network, data exchange must be time critical for minimizing the risk of car accidents on the network, unless a loss of human life and properties occurs, as unsafe data is exchanged in the network or messages are not delivered on the time for receiver vehicular node. For these purposes, long short term memory (LSTM), which is a type of recurrent neural network (RNN) is used that is very critical for problem solving in time serious data.

Secondly, MLP which is a type of neural networks is used for classifying and predicting our dataset. The implementation of deep learning is taking place on the semi-fog layer of the vehicular network for detecting malicious node communication and we are deploying deep learning on the fog layer of cloud computing (RSU) and on selected OBU of vehicles in case of roadside unit fails to work.

This work is efficient and effective for detecting internal and external communication in the vehicular network and minimizing the communication and computation delay directly implemented in cloud servers as well as minimizing the cost of using edge or fog servers for implementing deep learning detection. Distributing the deep learning by using RSU solves that only from some domain/range of data is checked whether the data is normal or malicious for sharing in the range and sends the malicious node information to the next RSU before the malicious nodes become a member of the next RSU and the RSU also share the node as malicious one for every vehicular node in the range and decrease the computational resource of identifying the malicious node after reaching every RSU. For training and testing our deep learning algorithm, we are using WSN-DS dataset by customizing it to become appropriate in our anomaly detection model scenario.

4 Chapter four

Dataset survey

4.1 Survey on vehicular network dataset and deep learning approach

Nowadays, IoT plays a great role in all aspects of human life by making the everyday activities of its users easy by providing remote access and controlling things, and exchanging data between those things by embedding sensors, software and other technologies using the internet. The Internet of vehicle is one branch of IoT that integrates the vehicular ad hoc network into the internet of things for creating large-scale vehicular networks.

In vehicular network, the communication is carried out through wireless communication and this communication is prone to different types of attacks. The purpose of securing communication on the internet of vehicles is to allow only correct information to be shared in the network and block abnormal data that leads to creating accidents in the vehicular network.

On the internet of vehicle where vehicles that connect to the network become very large and large amounts of data are disseminated on the network, it is difficult to differentiate normal data from abnormal or malicious data. There are different mechanisms of securing communication in the vehicular network for creating safety for the drivers and decreasing the occurrence of accidents by timely delivering genuine information between vehicles.

In a vehicular network, it is possible to secure communication of nodes by using authentication and other crypto analysis after they become a member of the network and it is possible only to differentiate external attacks and any node that has become a member of the network mimics as attacker without any challenge.

By using machine learning and deep learning, it's possible to detect either internal and external intrusions or anomalies in the vehicular network. Deep learning is a subset of machine learning and artificial intelligence that has higher detection accuracy than machine learning on large volumes of data. Many papers are conducted in anomaly detection of vehicular networks by using deep learning and they can achieve some detection accuracy by using different approaches. By considering some limitations of previous works by different scholars, this paper comes with anomaly based intrusion detection model on semi-fog of Internet of Vehicle services by using deep learning approach for detecting intrusion in the vehicular network. We are proposing a hybrid deep learning algorithm for achieving a high detection rate by using the advantage of the two deep learning together.

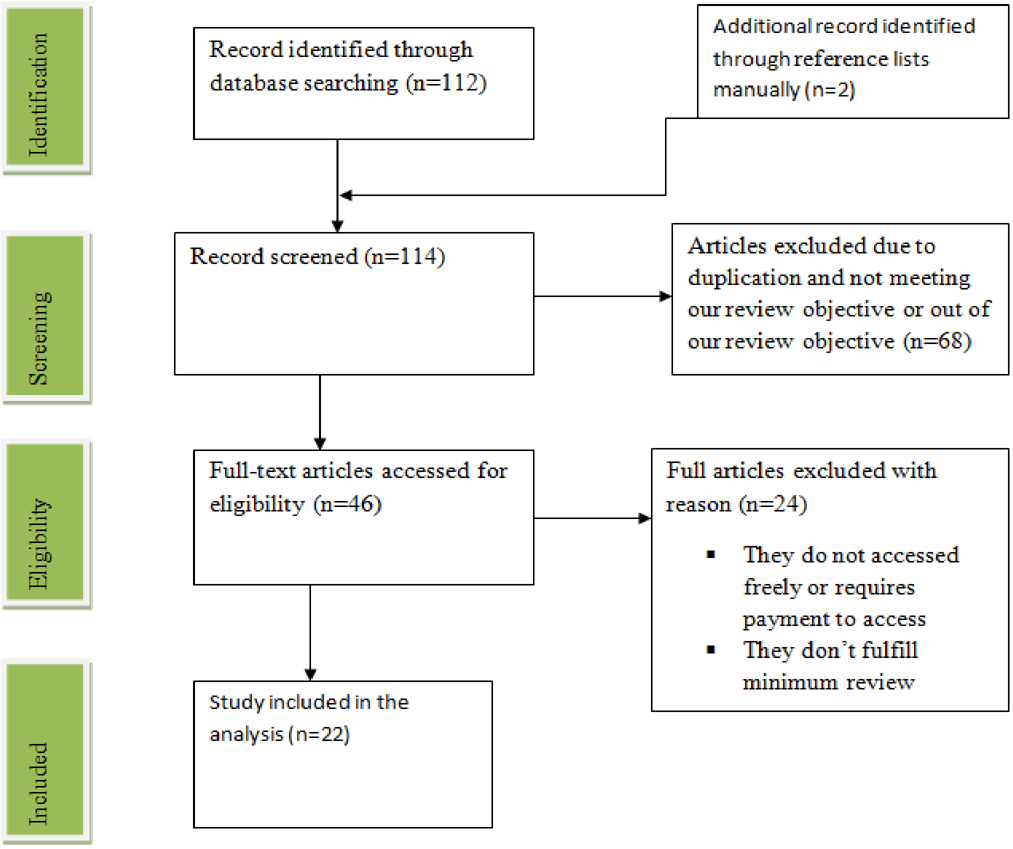
Identification
Record identified through database searching (n=112)
Additional record identified through reference lists manually (n=2)
Screening
Record screened (n=114)
Articles excluded due to duplication and not meeting our review objective or out of our review objective (n=68)
Eligibility
Full-text articles accessed for eligibility (n=46)
Full articles excluded with reason (n=24)
▪ They do not accessed freely or requires payment to access
▪ They don't fulfill minimum review
Included
Study included in the analysis (n=22)

Figure 6: Flow chart of the study selection and inclusion (n is number of papers at each for inclusion and exclusion)

4.2 Searching strategy and literature search

To understand and get enough ideas in the area of detecting anomalies/intrusion/misbehavior in the vehicular network by using deep learning and what approach they can follow for classifying their data and what type of dataset is used in their approach, we did a survey of previously done articles and journals by developing keywords for automatic searching from international databases repositories and systematic review is done. For automatically getting different articles and journals from different paper repositories, keywords in tables are used and after accessing the articles and journals, automatic review is made on the automatically gained information.

Literature search is made by using Google scholars and IEEE xplore advanced search paper database repositories. The searching approach was developed by using various keywords as shown in the table below and the method includes different articles that mostly focused on securing communication of internet of vehicles by using a deep learning approach. We are searching for recently published articles from Google scholar and IEEE Xplore in English language starting from the year 2018 to 2022.

Databases	Keywords
IEEE xplore Advance search	‘‘internet of vehicle’’ or ‘‘vehicular network’’ or ‘‘VANET’’ and ‘‘intrusion detection’’ or ‘‘anomaly detection’’ or ‘‘misbehavior detection’’ and ‘‘deep learning’’
Google scholar	‘‘internet of vehicle’’ or ‘‘vehicular network’’ or ‘‘VANET’’ and ‘‘intrusion detection’’ or ‘‘anomaly detection’’ or ‘‘misbehavior detection’’ and ‘‘deep learning’’
Manual search	Manual search based on citation and related articles

Table 2: Summary of the search strategy

4.3 Illegibility assessment

In the inclusion and eligibility of necessary articles and journals, different steps were performed for identifying and including the most important papers to be consisted in the review. For identifying the eligibility of each paper, the title, abstract, objective of the study and approach they used is reviewed. As a whole, studies were included if they: - (I) demonstrated and state securing the communication of vehicular network, (II) look in to or investigate vehicular nodes can share only normal data for minimizing car accident risk and loss of human life by timely delivering the message in the network and by using different deep learning approaches they can blocking malicious data from exchanged between vehicular nodes. (III) Creates safety for drivers, passengers and pedestrians by improving the way of secure information exchange in intelligent transportation system.

4.4 Data extraction and data analysis

Data extraction from different articles and journals was extracted by using thruuuserp analyzer application and an Excel database. The extracted data consists of title, author, publisher, year of publication, volume, of the articles and journals.

4.5 Survey result

A total of 114 articles were identified for inclusion and illegibility assessment and out of the total articles 68 articles excluded after title and abstract is reviewed. After exclusion is made based on some reason only 46 articles are become illegible for full article review and then only 22 articles are selected for review and met the last inclusion criteria. The included articles were published between 2018 and 2022.

4.5.1 Overview of the result

Out of 22 included papers, 7/22(31.82%) from IEEE, 2/22(9.1%) from Elsevier, 2/22(9.1%) from springer, 2/22(9.1%) from research gate, 2/22(9.1%) from Hindawi, 2/22(9.1%) from journal of MDPI, 1/22(4.545) from PLOS one, 1/22(4.545) from journal of IJCNA, 1/22(4.545) from journal of TCSST, 1/22(4.545) from semantic scholar and 1/22(4.545) from ACM digital library.

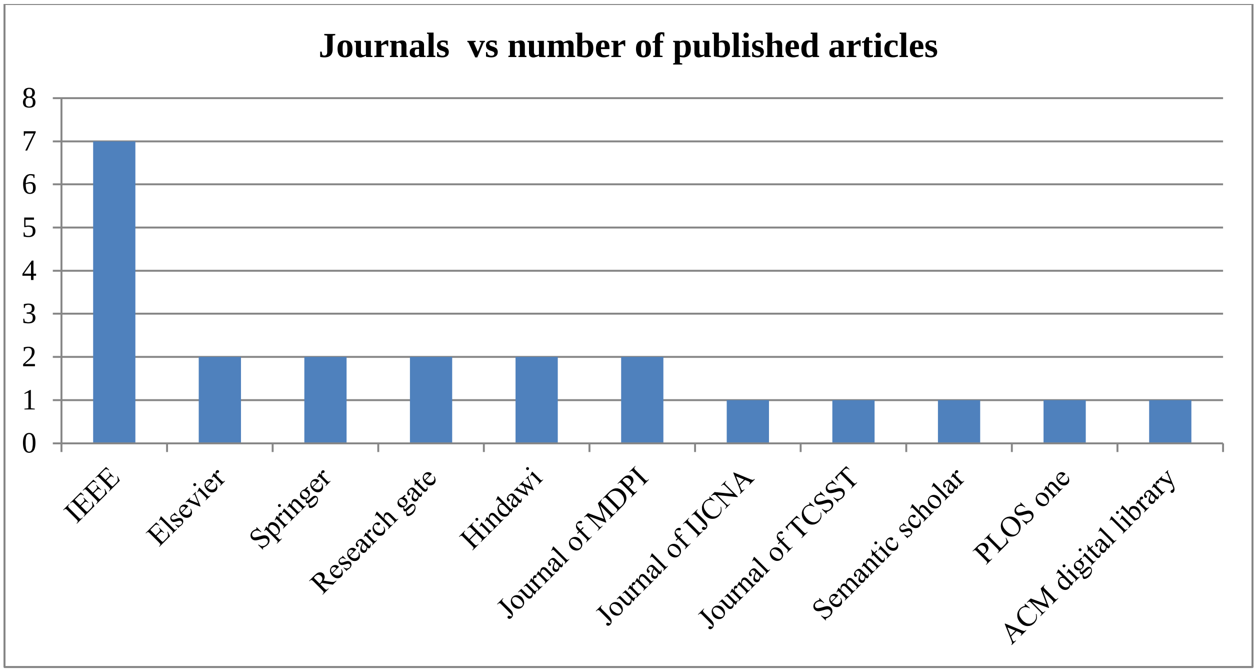

Figure 7: Summary of journal

From 22 included papers 7/22(31.82%), 6/22(27.272%), 4/22(18.18%), 3/22(13.64%) and 2/22(9.09%) are publish in 2022,2021,2020,2019 and 2018 respectively.

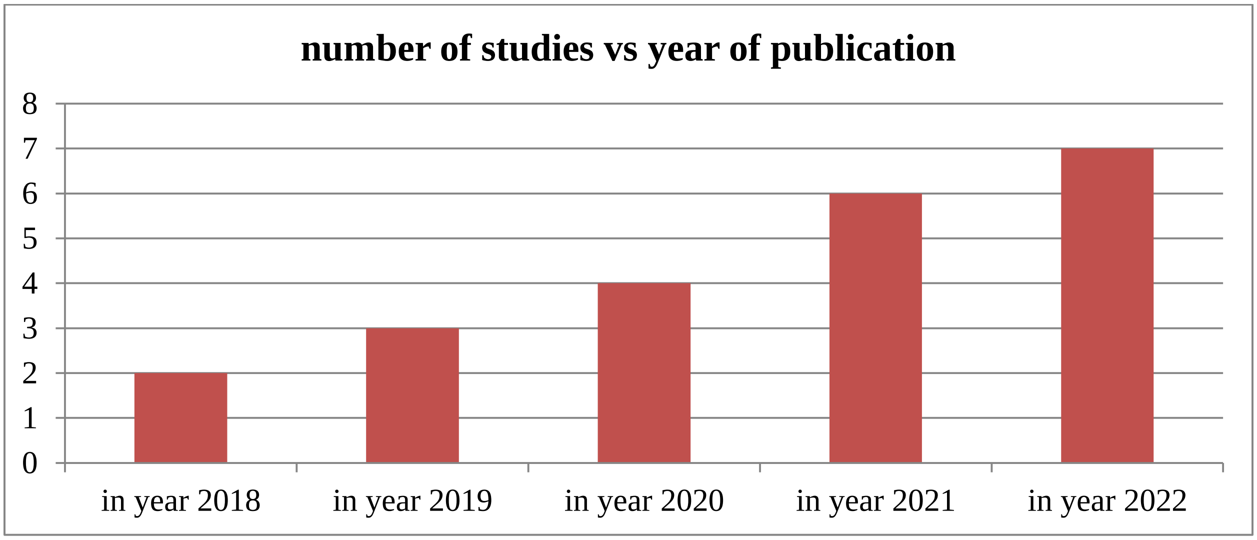

Figure 8: Summary of publication year

4.6 Deep learning based anomaly detection approach in vehicular network

<table>
<tr><th colspan="2">Approach</th><th>Dataset used</th><th>Number of articles</th><th>Over all dataset used</th></tr>
<tr><td rowspan="9">Hybrid two or more deep learning</td><td rowspan="3">CNN and LSTM</td><td rowspan="3">VeReMi extension dataset(2) and car hacking dataset(1), ISCX2012 IDS dataset(1)NSL-KDD UNSW-NB15 (1) and collect data by using OCTANE(1)</td><td rowspan="3">6/22(27.27%)</td><td>Car hacking dataset 4/22(18.18%)</td></tr>
<tr><td>VeReMi extension dataset 3/22(13.63)</td></tr>
<tr><td rowspan="3">Collecting their own dataset 6/22(27.27%)</td></tr>
<tr><td>Rule based and DNN</td><td>Their own dataset</td><td>1/22(4.54%)</td></tr>
<tr><td rowspan="2">Autoencoderand GAN</td><td rowspan="2">Car hacking datasets</td><td rowspan="2">1/22(4.54%)</td></tr>
<tr><td rowspan="3">KDD Cup99 dataset 3/22(13.63%)</td></tr>
<tr><td>LSTM and GRU</td><td>DDoS dataset and a car-hacking dataset</td><td>1/22(4.54%)</td></tr>
<tr><td rowspan="2">RNN, LSTM and RNN with attention mechanism</td><td rowspan="2">car-hacking dataset</td><td rowspan="2">1/22(4.54%)</td></tr>
<tr><td rowspan="2">NSL-KDD 2/22(9.09%)</td></tr>
<tr><td colspan="2">Stack LSTM</td><td>VeReMi dataset</td><td>1/22(4.54%)</td></tr>
</table>

DNN	KDD Cup99 and CICIDS 2018 dataset(1), their own dataset by collecting through using OCTANE(1) and VeReMi extension dataset(1)	3/22(13.63%)	UNSW-NB15 2/22(9.09%)
GAN	KDD Cup 99 and NSL-KDD(1) and their own dataset collecting by using OBD II port(1)	2/22(9.09%)	CICIDS2017 1/22(4.54%) WSN-DS 1/22(4.54%)
DBN	CICIDS 2017dataset	1/22(4.54%)	
ANN	WSN-DS	1/22	
xNN	CICIDS2019 dataset and UNSW-NB15 dataset	1/22(4.54%)	CICIDS2018 1/22(4.54%)
ANN	Collecting Their own dataset by using OBD II port	1/22(4.54%)	CICIDS2019

CNN and quantized deep CNN	Collect their own dataset (1) and car hacking dataset	2/22(9.09%)	1/22(4.54%)
MLP(multilayer perceptron)	KDD Cup1999	1/22(4.54%)	ISCX2012 IDS dataset 1/22(4.54%)

Table 3: Overview of dataset used in the survey

According to our survey the overall dataset used become above 100%, this is because of some paper uses more than 1 dataset in their training and testing on the deep learning they develop.

By consider our scenario for detecting anomaly in vehicular network by using deep learning algorithm that is employed on RSU and selected OBU, we are choosing WSN-DS that is the best way for deploying the deep learning on selected vehicular node in case of RSU fails. The data set consists of 374661 records with 19 features and has 4 types of attack and normal data. We are dropping 3 features that are not important to our scenario.

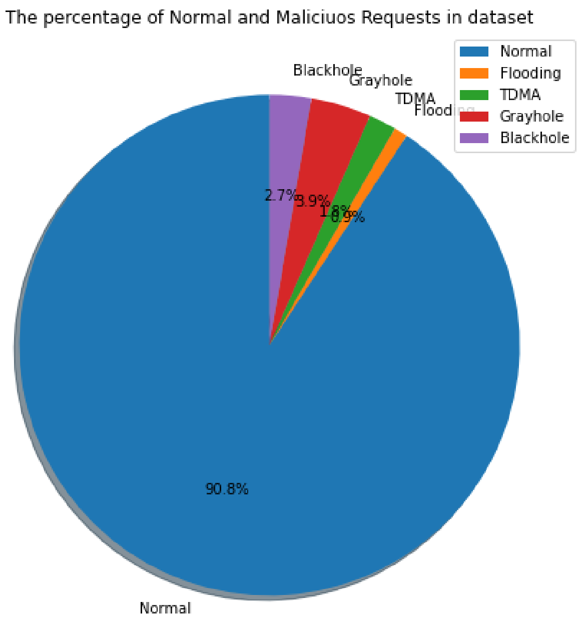

Figure 9: Representation of attacks in dataset in percent

5 Chapter five

Design of proposed solution

In this chapter, the proposed architecture for Anomaly based intrusion detection model on semi-fog of Internet of Vehicle services by using deep learning approach is detailed and discussed briefly. By considering most of the attacks existing in the vehicular network, we propose anomaly based intrusion detection model on semi-fog in the internet of vehicle for identifying malicious or misbehaving nodes in the network.

In vehicular network, attackers can get the advantage of releasing false information to the network by using or invading the onboard unit (OBU) and roadside Unit (RSU). In this scheme, we are designing the architecture of intrusion detection by using selected OBU and RSU of a vehicular network. In this anomaly detection model, we are developing and implementing semi-fog-based intrusion detection that exploits the RSU and selected OBU for detecting malicious data from exchanged in vehicular network.

In this work by considering a gap we stated in previous chapter we are developing and implementing anomaly based intrusion detection service on semi-fog of Internet of Vehicle services by using deep learning approach**.** In the modern internet of vehicular network communication is a vehicle to anything (V2X) and as the vehicle communicate to different components in the network and outside of the network, it increases the vulnerability of the network by different attack types.

For detecting both inter-vehicle communication and external communication attacks from the overall vehicular network system we are deploying the deep learning anomaly detection on RSU because every vehicular node communication is reached to RSU and in case of RSU fails and a vehicular node is out of the range of RSU we deploy the deep learning on a selected vehicular node.

In the vehicular network attacker can get the advantage of attacking both the OBU of the vehicle and RSU for exchanging malicious information in a vehicular network. Most papers done before this paper only implement their detection on OBU of vehicular nodes and as an RSU is invaded or controlled by a cyber-attacker every message that is sent by it is reached to every node in its domain and other RSUs then they can use this message as normal or genuine message and then an accident is occurred due to one RSU is attacked in a vehicular network.

For detecting both internal and external communication on resource constrains of OBU of vehicles it is difficult to deploy the heavy deep learning classifier on it. Only implementing deep learning on RSU also create some implication in a vehicular network. As the RSU that we implement the deep learning algorithm fails or stops working attacker can get the advantage of penetrating malicious information in the domain of fail RSU. By considering implementing the deep learning classifier only on specific OBU or RSU of vehicular networks for detecting malicious data exchange in vehicular network communication, we can put the deep learning on fog layer (RSU) and selected vehicular nodes.

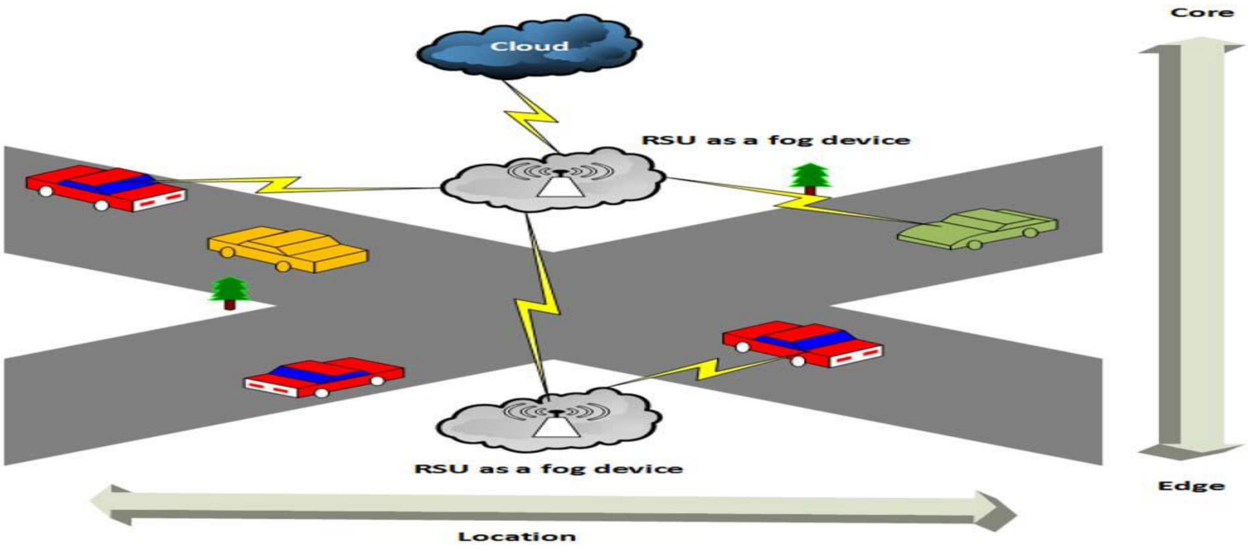

Figure 10: Cloud computing in IoV by Using RSU as fog device [101]

Cloud server: is a trust authority of the entire system and has powerful ability to compute and store massive amount of information. In this layer mainly TA, computing and storage resources are exists. The first responsibility of cloud layer is registering every vehicle and other service providing infrastructures by providing unique identity.

Fog layer: in cloud computing, some elements of the vehicular network infrastructure (such as RSU) are deployed near the edge of the network and they are interconnected to form a fog layer.

Vehicular nodes: each vehicle is equipped with an OBU, which shares some value information with other vehicles and RSU in the network (for example CAM and DENM).

5.1 RSU based intrusion detection

In this thesis we are using RSU as fog devices for implementing our deep learning algorithm due to the RSU has large computational and storage space. Unlike fog or edge server for deploying the deep learning algorithm RSU has delay free communication for time critical data.

Vehicular nodes can communicate with each other by using OBU for creating vehicle-to-vehicle communication (V2V) and also communicate with different vehicular network infrastructures named as a vehicle to infrastructure communication (V2I). Vehicle-to-roadside unit communication is one type of communication that takes place between the vehicle and roadside unit for exchanging safety messages for sharing in the network and also expanding the vehicular network. Every vehicle is communicated from one hop to the next vehicle and roadside unit for sharing information in the vehicular network. The roadside unit broadcasts information related to road safety, accident information, and road crowdedness to each vehicular node in their domain. Every CAM message broadcasted from OBU is reached the nearest RSU and the roadside unit broadcast that message to other RSU and vehicular node in its domain.

In the vehicular network every vehicular node broadcast sensitive message to one hop other vehicular nodes for creating communication between them and to RSU for creating a big vehicular network. Once an RSU receives a message from a vehicular node that is in its range and from another RSU it can broadcast that message to other roadside unit and other vehicular nodes in its domain. At the exchange of this critical message, an attacker can get the advantage of releasing malicious data in the network and creating an accident in the vehicular network. Attackers are mimicry as real vehicular nodes for releasing fake information in the network and selfish vehicular nodes also distribute false information for self-interest purposes. By considering attackers' interest in controlling the vehicular network for the distribution of fake and malicious data in the network, we are developing semi-fog-based anomaly detection by using a deep learning algorithm for detecting the distribution of malicious data in the network and for minimizing or mitigating the occurrences of accident and loss of life in a vehicular network.

This anomaly based intrusion detection system on semi-fog of Internet of Vehicle services scheme implement the deep learning on the fog layer (RSU) and selected OBU for detecting the overall internal and external communication of the vehicular network systems. In a vehicular network, the vehicular nodes move at high speed and dynamic position or change its position in a short period and only deploying deep learning on OBU is not efficient for detecting all attacks occurring at every place of IoV. The RSU is placed randomly near the road for creating a large network by exchanging information in a vehicular network. That RSU is stationary and it's an important infrastructure for creating communication between vehicular nodes and broadcasting sensitive CAM data to every node in its range and other RSUs.

Once the RSU receives vehicular node data it can put that data into a deep learning classifier module for identifying whether the data is normal or malicious. If the data is categorized as malicious one the RSU broadcast the information to every node in its range and other roadside units for not using any data that is sent from the attacker vehicle. One advantage of deploying deep learning on RSU is after data is identified as anomalous data in one RSU, the node becomes an attacker in the range of the checker RSU and it tells the vehicular node is an attacker one for other roadside units before the car joins the range of other RSU and the receiver RSU takes the node as attacker without applying the deep learning classifier and save computational time and resource for identifying the data as a normal or malicious one.

In a dense vehicular network that consists of a large number of vehicles and high volume of data is disseminated in the network and due to the limitation of resources for storing, computation and analyzing this big data on board unit of vehicle for identifying the normal and malicious data and the complexity of the deep learning for implementing on resource constraint OBU of vehicle, we are implementing the deep learning algorithm on semi fog layer(roadside unit) for detecting both in vehicle and external communication. This is done because the roadside unit has more resources than the onboard unit and is a more trusted vehicular network component and every vehicle can make communication with the roadside unit for creating a large vehicular network and also the deep learning classifier is implemented on selected OBU in case of RSU stop working.

5.2 Selecting vehicular node for implementing deep learning

For implementing the deep learning classifier on OBU on selected vehicular, we are using fixed cluster based cluster head selection algorithm that is used in[99]. In the paper the cluster head selection is used for authenticating purpose of disseminating data by aggregating to a single cluster head and sends to RSU for overcoming all nodes exchanged information to RSU and decrease the burden of RSU for authenticating each individual packet data sent from every vehicular node in the range and decrease the time and resource used for authenticating individual packet sent. In the paper they are using trust value (TVi), distance (Di) to cluster boundary and relative speed of vehicle in the cluster for fitness value and we are adding lane of vehicle for cluster formation. A vehicle become a member of a cluster if it travel in the same lane to other cluster member and then it also get a chance of becoming cluster head depending on fitness value. We are adding the lane for the purpose of vehicle that travel on the same lane has almost nearly similar speed and the selected cluster head function for a long time without changing the CH for cluster.

$F_{i=} W_1*TV_i+W_{2*}D_i-W_3|s_i-s_{avg}|$……………………………………………..(1)

W1, W2 and W3are weight denoting the relative importance of each component of the fitness function, where 0<=W1, W2, W3<=1 and W1+W2=W3=1. The highest the fitness value of a vehicle, a more suited it is to be selected as cluster head.

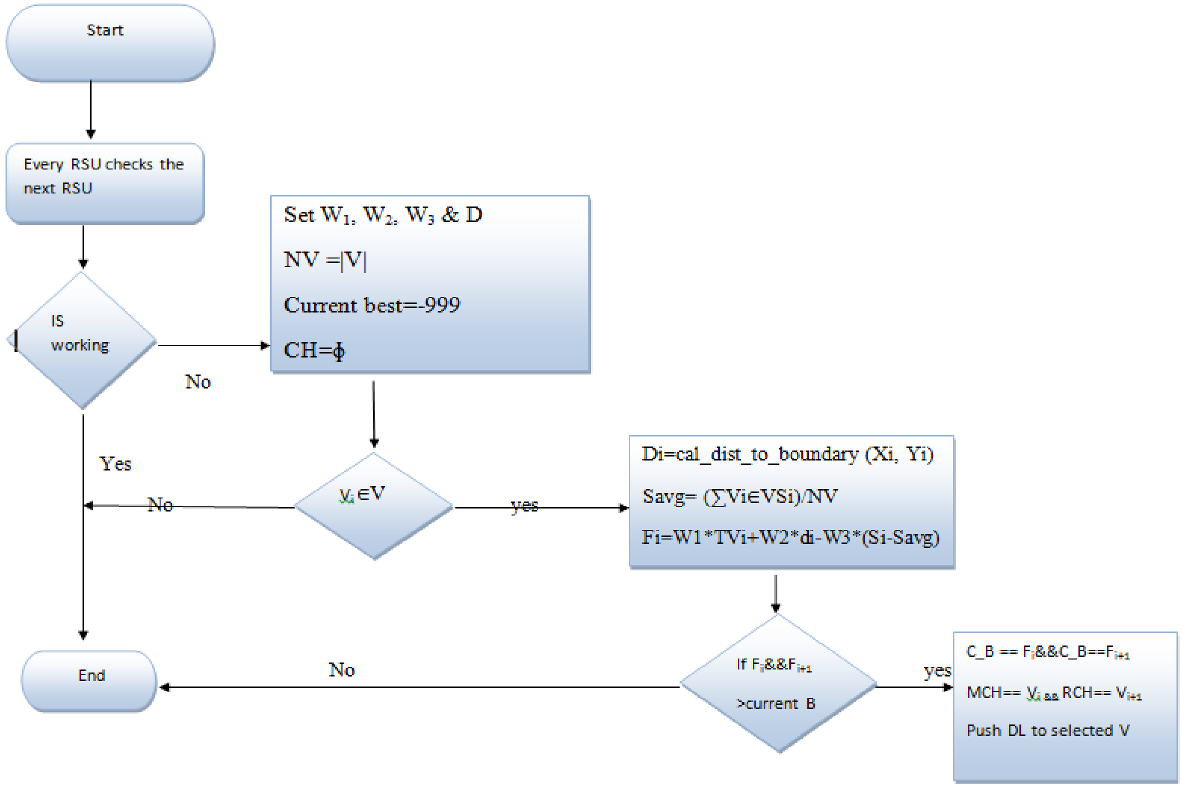
Start
Every RSU checks the next RSU
IS working
No
Set W_1, W_2, W_3 & D
NV =|V|
Current best=-999
CH=ϕ
Yes
V_i ∈V
No
yes
Di=cal_dist_to_boundary (Xi, Yi)
Savg= (∑Vi∈VSi)/NV
Fi=W1*TVi+W2*di-W3*(Si-Savg)
End
No
If F_i&&F_{i+1} >current B
yes
C_B == F_i&&C_B==F_{i+1}
MCH== V_i && RCH== V_{i+1}
Push DL to selected V

Figure 11: Flow chart of cluster head selection

Components of Fitness function are:

- Trust value (TVi) of a vehicle: the trust value of a vehicle is a metric that estimates how reliable the vehicle and trusted in the communication of vehicular network. Vehicle with high trust value are more reliable and suited to function as CH. In the case of this paper the trust level is calculated by past action of vehicle behavior like report of malicious behavior and in any reputation. For calculating fitness value we are assigning TV between 10 and 60 (10<=TV<=60) .
- Distance (d_i) to cluster boundary: this value is calculated based on the current position ($x_{i,}$ y_i) of vehicle $v_i$and the cluster boundary where that is ahead of the vehicle. If the value of D_i is higher, that means the vehicle is farther from the cluster boundary and is likely to remain in the current cluster for a long period of time and minimizes the overhead of new cluster head as the existing cluster head leaves the cluster boundary.
- Relative speed of a vehicle in the cluster: this metrics measures how the speed of a vehicle compares with the other vehicles in the cluster. A vehicle speed is similar to other neighboring vehicles in the cluster to be selected as cluster head, as it is more likely it will be travelling together with its cluster member for a long time. If the relative speed of a vehicle decreases the fitness function value increases and then it is selected as CH in the cluster.

  Relative speed of a vehicle in the cluster=$|S_i-S_{avg}|$
- Lane of the vehicle: we are choosing this component based on most of vehicles that travel on the same lane has almost similar car types and their speed become already not different and for formation of cluster it is important component one.

Illustration of cluster head selection by using example

The dashed line on the left hand side shows the cluster boundary, in our scenario it is the end of working RSU and the starting of the failed RSU and the vehicle speed from 25km/hr to 40km/hr. The below figure consists of vehicular trust value, speed and distance from the cluster boundary for the five cars.

S_{avg}= 32+40+29+34+37/5=34.4km/hr.

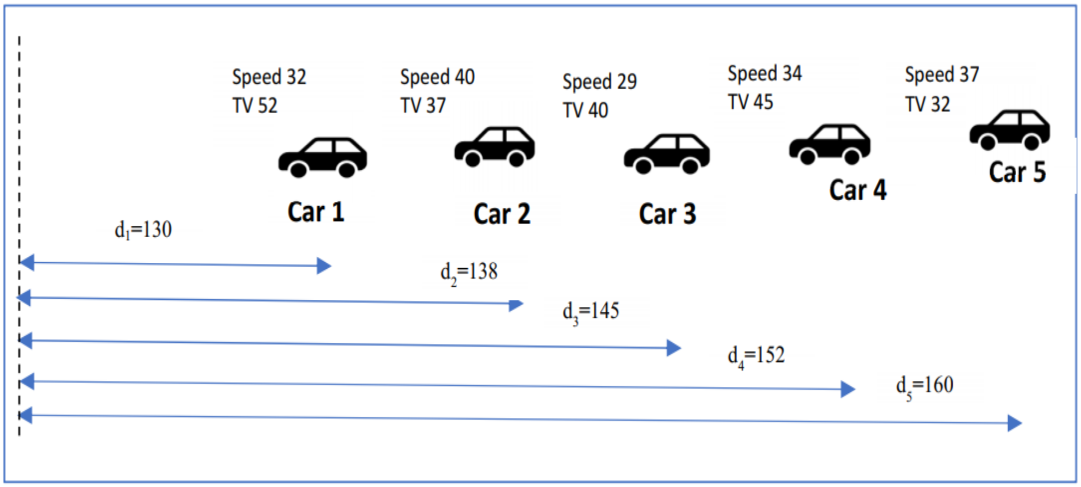

Figure 12: Status of nodes in cluster at time t1[99]

We are using $W_{1=}$0.4, W_2=0.4 and W_3=0.2

	TV_i	D_i from cluster boundary	Relative speed of vehicle in the cluster (|S_i-S_{avg}|)	Fitness value F_i=W1*TV_i+W2*D_i-W3*(|S_i-S_{avg}|)	Rank of becoming CH
Car1	52	130	2.4	72.32	4th
Car2	37	138	5.6	68.88	5th
Car3	40	145	5.4	72.92	3rd
Car4	45	152	0.4	78.72	1st(main CH)
Car5	32	160	2.6	76.28	2nd(reserve CH)

Table 4: Node attribute with fairness values

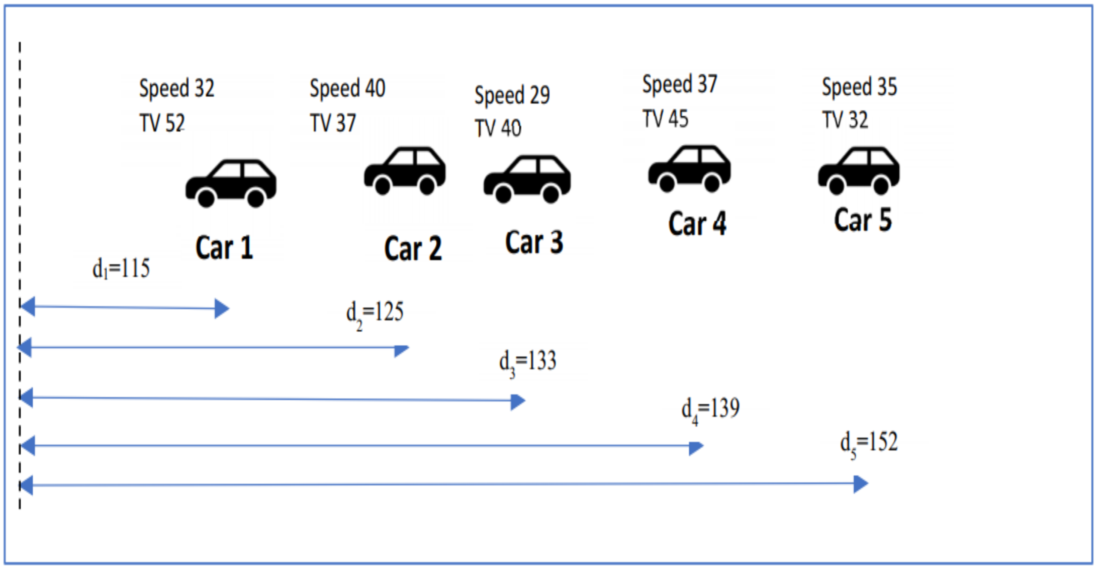

Figure 13: Status of nodes in a cluster at t2[99]

As the speed of vehicle and distance from cluster boundary changed from previous one we get another fitness values.

We are using the same Weights to the above example

Average speed of vehicles in cluster S_{avg}=32+40+29+37+35/5=34.6

	TV_i	D_i from cluster boundary	Relative speed of vehicle in the cluster ($\|S_i-S_{avg}\|$)	Fitness value $F_i=W1*TV_i+W2*D_i-W3*(\|S_i-S_{avg}\|)$	Rank of becoming CH
Car1	52	115	2.6	66.28	4th
Car2	37	125	5.4	63.72	5th
Car3	40	133	5.6	68.08	3rd
Car4	45	139	2.4	73.12	2nd(reserve CH)

Car5	32	152	0.4	73.52	1st(main CH)

Table 5: Node attributes with fairness values

In our work by considering fitness values of vehicle in the cluster the cluster member that has highest fitness value is selected as main CH and a vehicle that has the second highest fitness value as reserved CH. The reserved cluster head is selected in case of the main cluster head releasing the cluster boundary or some mechanical problem happens on the main CH and stopping on the road. The cluster head selection is made by road side unit.

Before cluster head selection, the RSU can check whether the next RSU is working or fail to work. If the next RSU is fail to work the checker RSU considers there is no one take responsibility of checking the exchanged information is normal or abnormal in that range and inform all vehicular node that travel in the position of failed RSU to form cluster and after that based on informed information all members under it sends hello message. After receiving the message it can select two cluster head as main cluster head and reserved cluster head based on fitness values. After selecting those two vehicles the RSU push the deep learning classifier on OBU of vehicles. The vehicle that is selected as CH receives all communication in the cluster and put the received data in the deep learning classifier module and then identify each data comes from neighboring vehicle is normal or malicious. If the main CH stop working due mechanical problem or leaving the cluster by increasing its speed, the reserved CH then become main cluster head then do what the main CH does before.

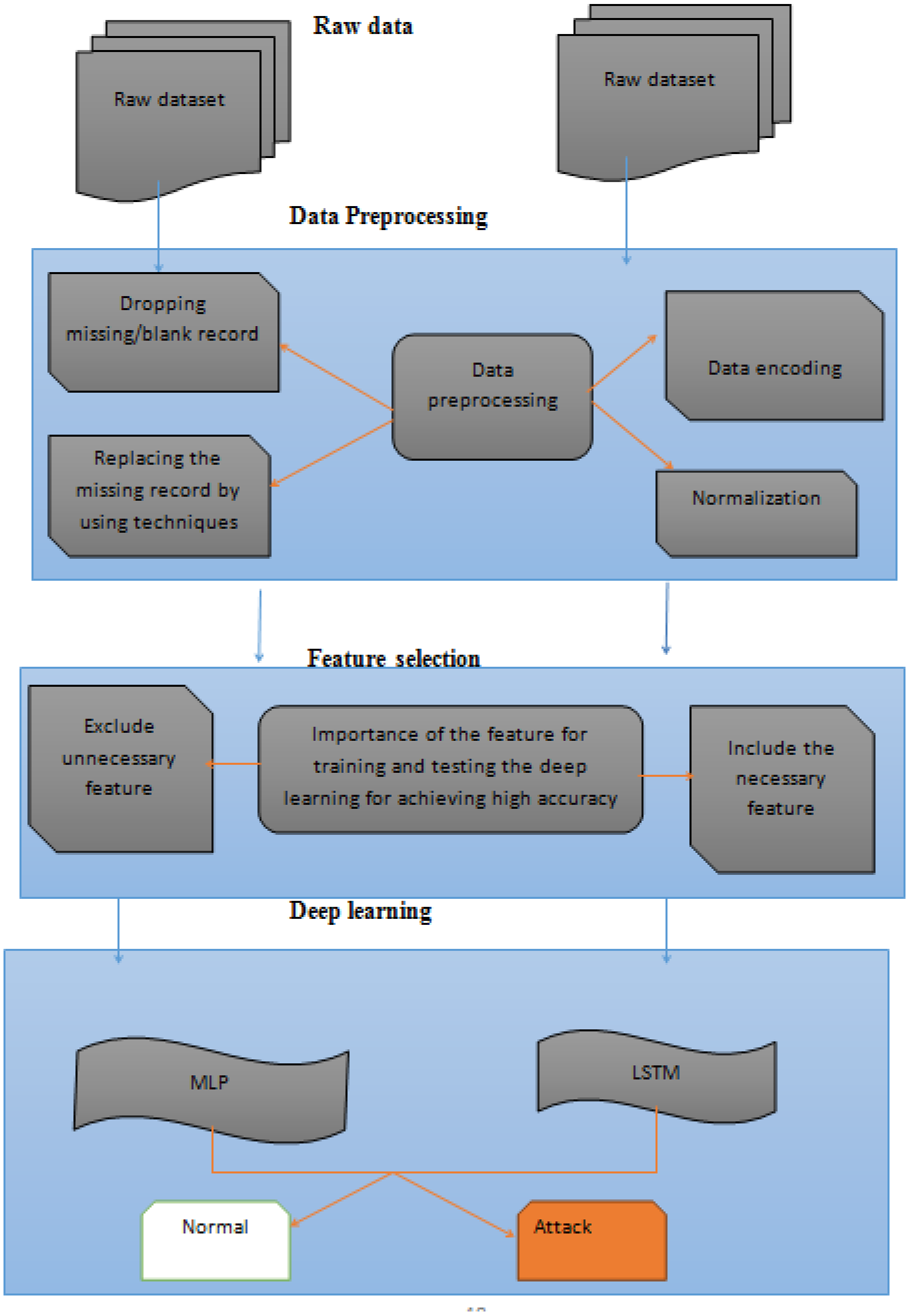
Raw data
Raw dataset
Raw dataset
Data Preprocessing
Dropping missing/blank record
Replacing the missing record by using techniques
Data preprocessing
Data encoding
Normalization
Feature selection
Exclude unnecessary feature
Importance of the feature for training and testing the deep learning for achieving high accuracy
Include the necessary feature
Deep learning
MLP
LSTM
Normal
Attack

Figure 14: The Proposed workflow

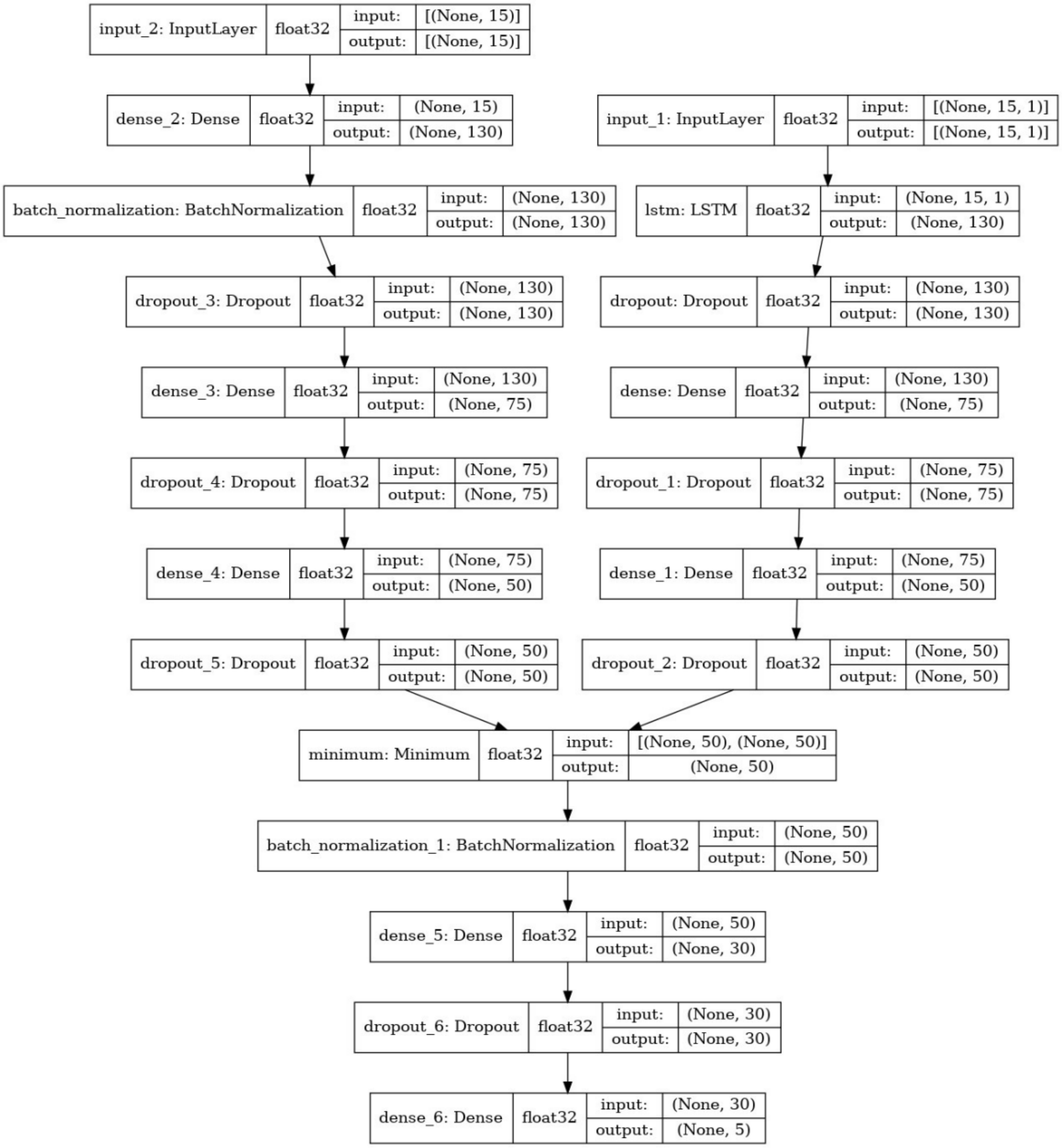
input_2: InputLayer | float32 | input: [(None, 15)] | output: [(None, 15)]
dense_2: Dense | float32 | input: (None, 15) | output: (None, 130)
input_1: InputLayer | float32 | input: [(None, 15, 1)] | output: [(None, 15, 1)]
batch_normalization: BatchNormalization | float32 | input: (None, 130) | output: (None, 130)
lstm: LSTM | float32 | input: (None, 15, 1) | output: (None, 130)
dropout_3: Dropout | float32 | input: (None, 130) | output: (None, 130)
dropout: Dropout | float32 | input: (None, 130) | output: (None, 130)
dense_3: Dense | float32 | input: (None, 130) | output: (None, 75)
dense: Dense | float32 | input: (None, 130) | output: (None, 75)
dropout_4: Dropout | float32 | input: (None, 75) | output: (None, 75)
dropout_1: Dropout | float32 | input: (None, 75) | output: (None, 75)
dense_4: Dense | float32 | input: (None, 75) | output: (None, 50)
dense_1: Dense | float32 | input: (None, 75) | output: (None, 50)
dropout_5: Dropout | float32 | input: (None, 50) | output: (None, 50)
dropout_2: Dropout | float32 | input: (None, 50) | output: (None, 50)
minimum: Minimum | float32 | input: [(None, 50), (None, 50)] | output: (None, 50)
batch_normalization_1: BatchNormalization | float32 | input: (None, 50) | output: (None, 50)
dense_5: Dense | float32 | input: (None, 50) | output: (None, 30)
dropout_6: Dropout | float32 | input: (None, 30) | output: (None, 30)
dense_6: Dense | float32 | input: (None, 30) | output: (None, 5)

Figure 15: Deep learning Model

5.3 Data preprocessing, encoding and normalization

In data preprocessing phase we are firstly identify blank records from the dataset and we are giving high attention to those missing records because in some cases those missing data value can decrease the result of deep learning algorithm. After we are identifying the missing record we are try to replace by different mechanisms but it is become impossible to give values and our last option is deleting those null values.

In encoding phase all symbolic and textual data attributes are convert to numeric values and the encoding begin at one and increase one with each feature for each symbolic and textual attributes.

Data normalization is applied because some parameters vary in different range and to eliminate this range difference in parameters standard method data normalization is applied. Its purpose is to standardize the data of different dimension and units to solve the difference between the data indicator. After normalization, different piece of data are at the same level, which is convenient for comprehensive comparative evaluation because of after data normalization is applied on the data the feature of data values are scaled in the range between 0.0 to 1.0. Unless we are normalizing the features of data our prediction is not accurate due to the difference between the two values in the experiment.

From many alternatives of normalization techniques we are choosing Min-Max due to its simplicity and it is necessary to use the unified fixed minimum and maximum values.

N=n-min/max-min ……………………………………………(2)

Where N and n is new and old data values

5.4 Feature selection

The main features are selected for achieving high accuracy, the detection classifier's precision and decreasing number of false alarm. Out of 19 features in the dataset we are selecting only 16 features for customizing it to our scenarios by dropping three features that is unnecessary for our result and they are excluded in training and testing of the dataset in our deep learning classifier algorithm.

5.5 Classification and prediction of attack types

In this work the deep learning algorithm trained to classifies the raw data in to normal and malicious data packet and then by using regression the classified data is predicted as specific attack types. In the first stage the preprocessed data is trained to classify the raw data into two groups: normal and abnormal or malicious. In the second stage the classified data can be predicted as specific attack types consisted in dataset as flooding, black hole, gray hole and scheduling attacks.

6 Chapter six

Implementation and performance evaluation

6.1 Overview

For training and testing our deep learning model, we use a WSN-DS dataset that is release publicly. The dataset consists of imbalanced dataset distribution. Most percent is occupied by normal data (90%) and this not good for efficient classification and creates overfitting problem. To overcome this problem synthetic minority oversampling technique (SMOTE) is used, and then we are comparing the result with other paper that uses similar dataset. The result that we got our deep learning shows that our deep learning model efficiently classifies the dataset.

6.2 Tools used

- Kaggle is used to train and test our model
- Jupyter Note book for writing and editing our code
- Python programming language

6.3 Dataset description and data balancing

6.3.1 Dataset description

We are using WSN-DS for training and testing our deep learning classifier algorithm. This dataset is public dataset that is used for research purpose which consists of around 19 features. We are choosing this dataset because it fulfills our requirement of detecting anomaly in vehicular network by selecting cluster head in case of RSU fails or stop to do the deep learning classification. The dataset feature is mostly convenient for implementing anomaly detection by selecting cluster head and our scenario behalf of dependent on cluster formation and cluster head selection where RSU not functioning. The dataset consists of 374661 data where four types of attack and normal data. We are using 70% for training and 30% for testing from overall dataset.

Attack type	Number of record	Percentage
Blackhole	10049	2.682%
Grayhole	14596	3.896%
Scheduling	6638	1.772%

Flooding	3312	0.884%
Normal	340066	90.766%
Total	374661	100%

Table 6: Dataset distribution

Attack type	Number of record	Percentage
Blackhole	7005	2.671%
Grayhole	10159	3.873%
Scheduling	4626	1.764%
Flooding	2287	0.872%
Normal	238185	90.820%
Total	262262	100%

Table 7: Training dataset distribution

Attack type	Number of record	Percentage
Blackhole	3044	2.708%
Grayhole	4437	3.948%
Scheduling	2012	1.790%
Flooding	1025	0.912%
Normal	101881	90.642%
Total	112399	100%

Table 8: Testing dataset distribution

6.3.2 Dataset balancing

The dataset that we used to train and test our deep learning consist of imbalanced data distribution and this leads overfitting problem in our deep learning algorithm. The normal data takes the most percent of the overall data and this is not necessary to achieving best accuracy of classifying the dataset and detecting the anomalous data. Some machine learning techniques are bias towards majority class, and they tend to ignore the minority class. To overcome the overfitting problem we are applying synthetic minority oversampling technique (SMOTE). SMOTE is a data augmentation algorithm that creates synthetic data points from raw data and it has the advantage of not creating duplicate data points, but rather synthetic data points that differ slightly from the original data points. SMOTE synthesis new minority instances between existing minority instances. Creating balanced data is done by applying duplicating the data points of minority class or under sampling the majority class. In our case we are applying increasing the minority class by duplicating their data points. After applying the technique we get the following dataset distribution and the classification and prediction of each anomaly become more efficient.

Attack type	Number of records	Percentage
Blackhole	238216	20.02%
Grayhole	238255	20.03%
Flooding	237878	19.96%
Scheduling /TDMA	237673	19.97%
Normal	238209	20.02%
Total	1190231	100%

Table 9: Training dataset distribution after SMOTE

Attack type	Number of records	Percentage
Blackhole	101850	19.97%
Grayhole	101811	19.96%

Flooding	102393	20.07%
Scheduling	102188	20.03%
Normal	101857	19.97%
Total	510099	100%

Table 10: Testing dataset distribution after SMOTE

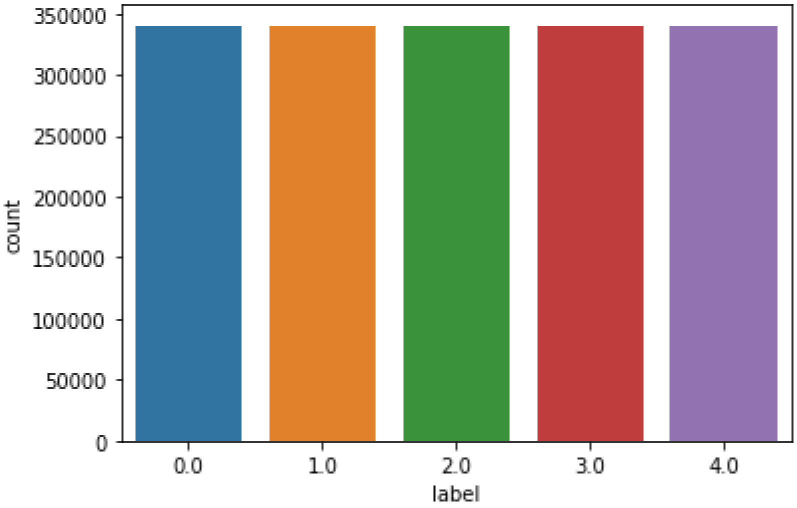

Figure 16: Dataset representation after SMOTE

6.4 Dataset features

WSN-DS has 19 features and out of this features we are customized it to 16 features based on our scenario by dropping three features.

Feature name	Description
ID	a unique ID to distinguish the node
Time	the current simulation time of the node

Is CH?	A flag to distinguish whether the node is CH with value 1 or normal node with value 0.
Who CH?	The ID of the CH in the current round
Distance to CH	the distance between the node and its CH in the current round
ADV_CH send	The number of advertise CH's broadcast messages sent to the nodes.
ADV_CH receives	the number of advertise CH messages received from CHs
Join_REQ send	the number of join request messages sent by the nodes to the CH
Join_REQ receive	The number of join request messages received by the CH from the nodes.
ADV_SCH send	The number of advertise TDMA schedule broadcast messages sent to the nodes.
ADV_SCH receives	The number of TDMA schedule messages received from CHs. Rank: the order of this node within the TDMA schedule.
Rank	The order of the node within the TDMA schedule
Data sent	the number of data packets sent from a node to its CH.
Data received	the number of data packets received from CH

Send Code	The cluster sending code.
Label / attack type	Type of the node. It is a class of five possible values, namely, Blackhole, Grayhole, Flooding, and Scheduling, in addition to normal, if the node is not an attacker.

Table 11: dataset features

6.5 Data preprocessing and training

Data preprocessing or data cleansing is the major tasks in modeling and training data in deep neural network for increasing the anomaly detection accuracy. For preparing the data is accurate and consistent for feeding it in deep neural network classifier algorithm the preprocessing processes is important one. Duplicated data is removed from data set and then after data encoding and normalization is applied for preparing quality data that can provide good detection accuracy in training and testing in deep neural network algorithm.

The preprocessed data is fed in to the deep neural network model that is separated in to two parts MLP and LSTM. The input layer has 15 neurons with ReLu activation function, in hidden layer there is many dense layer that takes an input and convert it to outputs and also there is dropout between different neurons line connection for overcoming overfitting the model withReLu activation function. Lastly the dense layer that comes from the two deep neural networks is concatenated in to one and then also passes a dense layer then gives an output layer with 5 neurons with softmax activation function.

Parameter	Value
Activation function	ReLU
Optimizer	Adam
Epochs	15
Loss	Sparse Categorical cross entropy

Batch size	64

Table 12: Parameter setting

6.6 Experiment and result

In this section we discuss experiment results achieved in the processes of conducting this paper.

Accuracy: Accuracy is a metric that generally describes how the model performs across all classes. It is calculated as the ratio between the numbers of correct predictions to the total number of predictions.

$$Accuracy = \frac{True_{positive} + True_{negative}}{True_{positive} + True_{negative} + False_{positive} + False_{negative}}$$

--------------------- (6.6.1)

Precision: The precision is calculated as the ratio between the number of *Positive* samples correctly classified to the total number of samples classified as *Positive* (either correctly or incorrectly).

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$= \frac{True\ Positive}{Total\ Predicted\ Positive}$$

------ (6.6.2)

Recall: The recall is calculated as the ratio between the numbers of *Positive* samples correctly classified as *Positive* to the total number of *Positive* samples. The recall measures the model's ability to detect *Positive* samples. The higher the recall, the more positive samples detected.

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$= \frac{True\ Positive}{Total\ Actual\ Positive} \qquad \text{---------- (6.6.3)}$$

F1 Score:

$$F1 = 2 \times \frac{Precision * Recall}{Precision + Recall} \qquad \text{--------------- (6.6.4)}$$

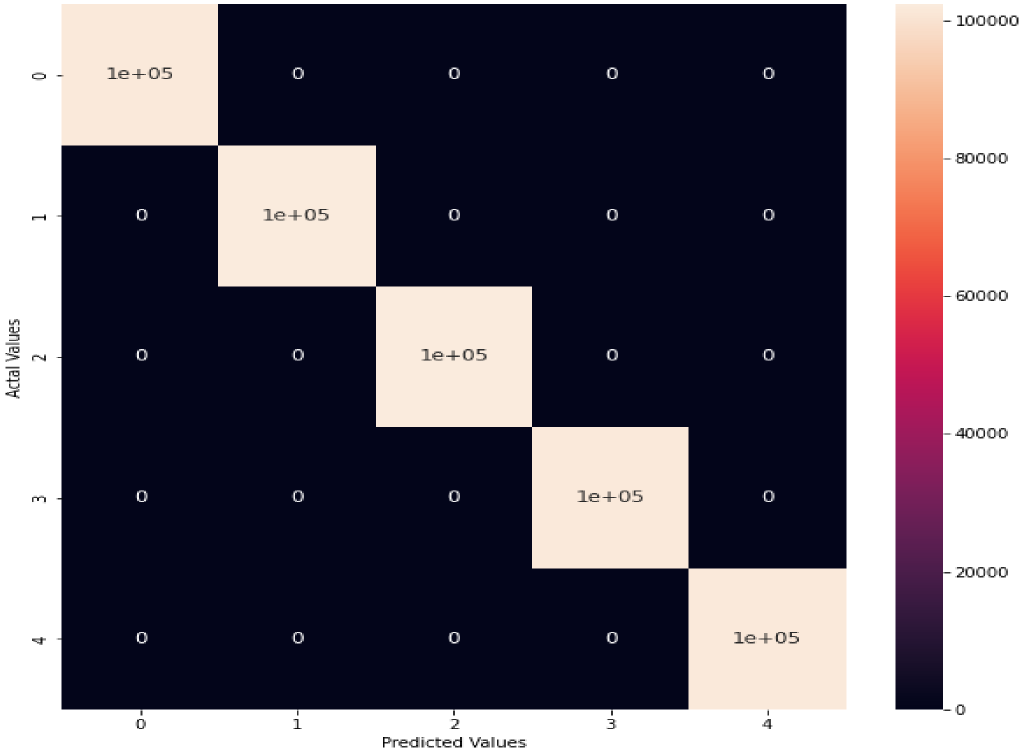

Figure 17: Confusion matrix

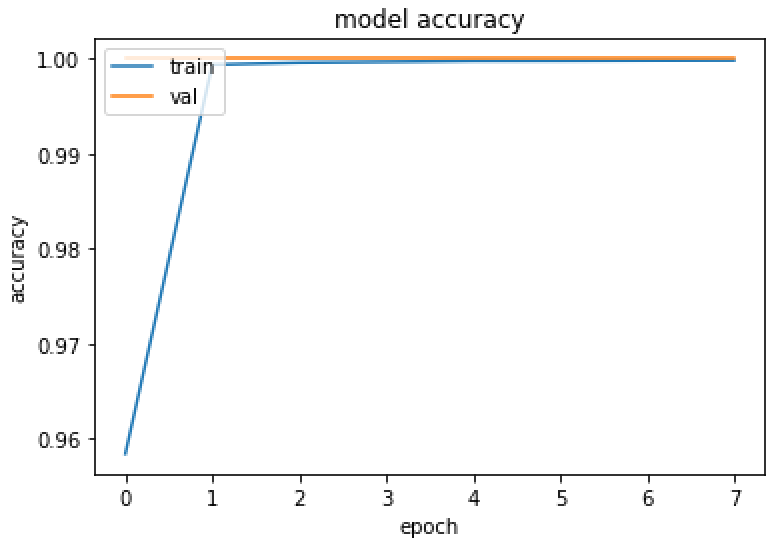

Figure 18: Accuracy to epoch graph after SMOTE

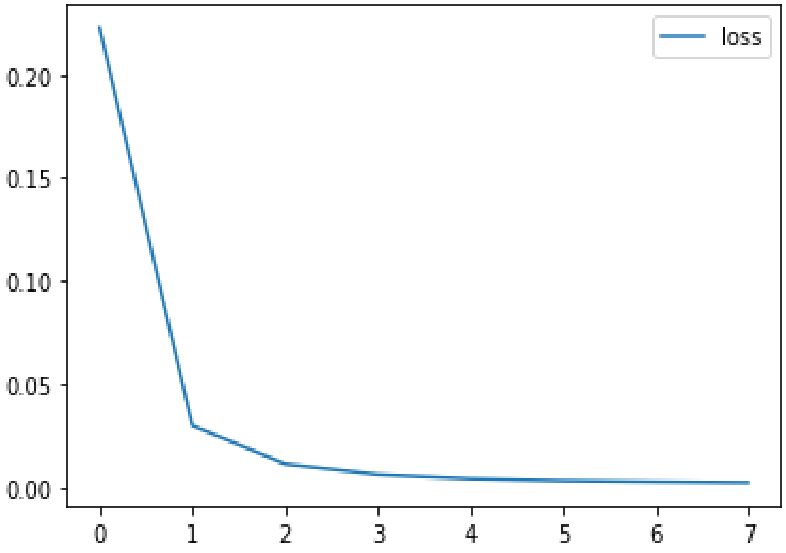

Figure 19: Loss graph after SMOTE

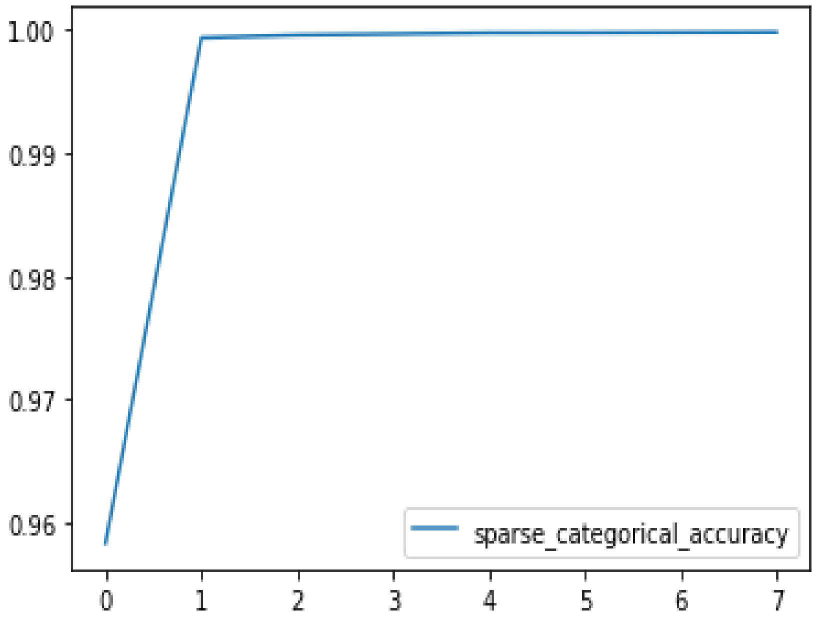

Figure 20: Accuracy graph after SMOTE

Attack type	After SMOTE results			
	Accuracy	Precision	Recall	F1-Score
Flooding	99.1%	100%	100%	100%
TDMA	98.8%	100%	63%	78.9%
grayhole	99.3%	85%	100%	92.7%
Blackhole	99.6%	100%	98%	99.1%

Table 13: Accuracy result after SMOTE

Comparison of results

Accuracy results of MLP (multilayer perceptron) and the hybrid of MLP with LSTM

To validate the hybrid model achieve best classification and identification of each attack types inside dataset ,we train and test the dataset by using single neural network MLP and we got 90.71% of accuracy and after hybriding it with LSTM we achieve 99.60% of accuracy.

Neural network model	Accuracy result
MLP	70.71%
Hybrid (MLP and LSTM)	99.80%

Table 14: comparison of MLP and MLP-LSTM

The above table shows our deep learning model achieves different results as we use single deep learning algorithm (multi-layer perceptron) and as we hybrid it with LSTM (LSTM-MLP). The hybrid approach performs effectively classifying our dataset and it prove that hybrid approach provide high accuracy.

Accuracy result Comparison of this work with another one with the same dataset

Paper and approach	Attack types			
	Flooding	Scheduling	Grayhole	blackhole
ANN [100]	99.4%	92.20%	75.6%	92.8%
MLP and LSTM(this work)	99.1%	99.60%	99.3%	99.6%

Table 15: Accuracy comparison of this work with other paper that uses same dataset

In the above table we are comparing the accuracy of detecting each attack types of our result with other work result that uses same dataset and our deep learning model detects the dataset effectively and we got best result on three attack types.

7 Chapter seven

Conclusion, contribution and future work

7.1 Conclusion

Internet of vehicle plays a vital role in the communicating safety message for safeguarding the driver, passengers, the vehicle itself in global level. Recently millions of people lose their life on the road due to accident. This accident occurs because of more sensitive messages doesn't deliver to the receiver vehicle on time due to selfish vehicle flooding unnecessary message to the network for creating congestion and attackers release false message in the network. As new vehicular node that joins vehicular network increase and more data is disseminate in the network, it is difficult to know message comes from the neighboring vehicle is genuine to use in the network or not.

In internet of vehicle the communication is not limited to vehicle to vehicle and vehicle to other vehicular network infrastructure and the communication is vehicle to anything, then it is possible for attacker to get a hole to access and penetrate fake or malicious data in the network.

For overcoming the problem of malicious data is disseminated in vehicular network, in this paper we tried to develop fog based anomaly detection model in internet of vehicle service by using deep learning algorithm. We are deploying the deep neural network classifier on RSU and selected OBU of vehicular node in case of RSU fail. Vehicle is selected as main CH and reserved CH from the cluster based on fitness value and the selection is made by the working RSU after checking the next RSU is fail. All vehicle travel toward the failed RSU creates cluster based on their lane and then the RSU select two vehicles as main and reserved CH and then push the deep learning neural network classifier on the selected vehicular nodes. The main CH accept all data in the cluster and check whether the data is normal and abnormal by using the deep learning classifier module that is pushed from RSU. The reserved CH serve as CH if only the main CH stopping due to mechanical problem or leaves the cluster by increasing its speed.

7.2 Contribution

The contribution of this work is designing and implementing deep learning algorithm that detects malicious data before disseminating in vehicular network by deploying the deep learning neural network classifier on:

- RSU – RSU as fog device is used for implementing the deep learning classifier which minimizes communication delay of directly implementing our deep learning classifier on edge server and the trusted RSU as bridge between the vehicles and edge server
- In addition the deep learning classifier is deployed on OBU of selected vehicles as cluster head in case of RSU fails.

7.3 Future work

Even if this work achieves its objective of detecting dissemination of anomalous data in internet of vehicle, it needs to be improved for the future. The following are some lists that we recommend as future work and someone else continue for future:

- This work is only limited to detect four different types of DoS attacks, for future they will add another attack types.
- We are using a dataset that is most convenient to wireless sensor network and IoT, for future they will to use dataset that is convenient for vehicular network.

References

[1] S. Latif *et al.*, "Multicriteria Based Next Forwarder Selection for Data Dissemination in Vehicular Ad Hoc Networks Using Analytical Network Process," *Mathematical Problems in Engineering*, vol. 2017, pp. 1–18, 2017.

[2] "Injuries and violence." https://www.who.int/news-room/fact-sheets/detail/injuries-and-violence (accessed Dec. 04, 2022).

[3] "Dedicated Short Range Communications (DSRC) Service," *Federal Communications Commission*, Apr. 20, 2016. https://www.fcc.gov/wireless/bureau-divisions/mobility-division/dedicated-short-range-communications-dsrc-service (accessed Dec. 04, 2022).

[4] H. Hartenstein and K. P. Laberteaux, Eds., *VANET: Vehicular Applications and Inter-Networking Technologies*, 1st ed. Wiley, 2010. doi: 10.1002/9780470740637.

[5] A. H. Sodhro *et al.*, "Towards 5G-Enabled Self Adaptive Green and Reliable Communication in Intelligent Transportation System," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 8, pp. 5223–5231, Aug. 2021, doi: 10.1109/TITS.2020.3019227.

[6] A. H. Sodhro, J. J. P. C. Rodrigues, S. Pirbhulal, N. Zahid, A. R. L. de Macedo, and V. H. C. de Albuquerque, "Link Optimization in Software Defined IoV Driven Autonomous Transportation System," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3511–3520, Jun. 2021, doi: 10.1109/TITS.2020.2973878.

[7] "Deep Learning Specialization." https://www.deeplearning.ai/courses/deep-learning-specialization/ (accessed Dec. 04, 2022).

[8] J. Li, Z. Zhao, R. Li, and H. Zhang, "AI-Based Two-Stage Intrusion Detection for Software Defined IoT Networks," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2093–2102, Apr. 2019, doi: 10.1109/JIOT.2018.2883344.

[9] G. Liu and J. Zhang, "CNID: Research of Network Intrusion Detection Based on Convolutional Neural Network," *Discrete Dynamics in Nature and Society*, vol. 2020, p. e4705982, May 2020, doi: 10.1155/2020/4705982.

[10] S. T. Mehedi, A. Anwar, Z. Rahman, and K. Ahmed, "Deep Transfer Learning Based Intrusion Detection System for Electric Vehicular Networks," *Sensors*, vol. 21, no. 14, p. 4736, Jul. 2021, doi: 10.3390/s21144736.

[11] T. Alladi, A. Agrawal, B. Gera, V. Chamola, B. Sikdar, and M. Guizani, "Deep Neural Networks for Securing IoT Enabled Vehicular Ad-Hoc Networks," in *ICC 2021 - IEEE International Conference on Communications*, Jun. 2021, pp. 1–6. doi: 10.1109/ICC42927.2021.9500823.

[12] L. Nie, Z. Ning, X. Wang, X. Hu, J. Cheng, and Y. Li, "Data-Driven Intrusion Detection for Intelligent Internet of Vehicles: A Deep Convolutional Neural Network-Based Method," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2219–2230, Oct. 2020, doi: 10.1109/TNSE.2020.2990984.

[13] P. Yurkovich, "RSU-Based Intrusion Detection and Autonomous Intersection Response Systems," p. 89.

[14] S. Gyawali, Y. Qian, and R. Q. Hu, "Machine Learning and Reputation Based Misbehavior Detection in Vehicular Communication Networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 8871–8885, Aug. 2020, doi: 10.1109/TVT.2020.2996620.

[15] H. Grover, T. Alladi, V. Chamola, D. Singh, and K.-K. R. Choo, "Edge Computing and Deep Learning Enabled Secure Multitier Network for Internet of Vehicles," *IEEE Internet of Things Journal*, vol. 8, no. 19, pp. 14787–14796, Oct. 2021, doi: 10.1109/JIOT.2021.3071362.

[16] T. Alladi, V. Kohli, V. Chamola, and F. R. Yu, "Securing the Internet of Vehicles: A Deep Learning-Based Classification Framework," *IEEE Networking Letters*, vol. 3, no. 2, pp. 94–97, Jun. 2021, doi: 10.1109/LNET.2021.3058292.

[17] B. C. Csáji, Z. Kemény, G. Pedone, A. Kuti, and J. Váncza, "Wireless Multi-Sensor Networks for Smart Cities: A Prototype System With Statistical Data Analysis," *IEEE Sensors Journal*, vol. 17, no. 23, pp. 7667–7676, Dec. 2017, doi: 10.1109/JSEN.2017.2736785.

[18] L. Khoukhi, H. Xiong, S. Kumari, and N. Puech, “The Internet of vehicles and smart cities,” *Ann. Telecommun.*, vol. 76, no. 9, pp. 545–546, Oct. 2021, doi: 10.1007/s12243-021-00891-7.

[19] L.-M. Ang, K. P. Seng, G. K. Ijemaru, and A. M. Zungeru, “Deployment of IoV for Smart Cities: Applications, Architecture, and Challenges,” *IEEE Access*, vol. 7, pp. 6473–6492, 2019, doi: 10.1109/ACCESS.2018.2887076.

[20] Y. Xie *et al.*, “STM32-based vehicle data acquisition system for Internet-of-Vehicles,” in *2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS)*, May 2017, pp. 895–898. doi: 10.1109/ICIS.2017.7960119.

[21] J. Chen *et al.*, “Service-Oriented Dynamic Connection Management for Software-Defined Internet of Vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 10, pp. 2826–2837, Oct. 2017, doi: 10.1109/TITS.2017.2705978.

[22] “Full article: A seven-layered model architecture for Internet of Vehicles.” https://www.tandfonline.com/doi/full/10.1080/24751839.2017.1295601 (accessed Dec. 03, 2022).

[23] M. Ru, S. Yin, and Z. Qu, “Power and Spectrum Allocation in D2D Networks Based on Coloring and Chaos Genetic Algorithm,” *Procedia Computer Science*, vol. 107, pp. 183–189, 2017, doi: 10.1016/j.procs.2017.03.076.

[24] D. Lin, Y. Tang, and A. V. Vasilakos, “User-Priority-Based Power Control in D2D Networks for Mobile Health,” *IEEE Systems Journal*, vol. 12, no. 4, pp. 3142–3150, Dec. 2018, doi: 10.1109/JSYST.2017.2673870.

[25] C. Spelta, V. Manzoni, A. Corti, A. Goggi, and S. M. Savaresi, “Smartphone-Based Vehicle-to-Driver/Environment Interaction System for Motorcycles,” *IEEE Embedded Systems Letters*, vol. 2, no. 2, pp. 39–42, Jun. 2010, doi: 10.1109/LES.2010.2052019.

[26] N. Salameh, G. Challita, S. Mousset, A. Bensrhair, and S. Ramaswamy, “Collaborative positioning and embedded multi-sensors fusion cooperation in advanced driver assistance system,” *Transportation Research Part C: Emerging Technologies*, vol. 29, pp. 197–213, Apr. 2013, doi: 10.1016/j.trc.2012.05.004.

[27] F. Chiti, R. Fantacci, Y. Gu, and Z. Han, “Content sharing in Internet of Vehicles: Two matching-based user-association approaches,” *Vehicular Communications*, vol. 8, pp. 35–44, Apr. 2017, doi: 10.1016/j.vehcom.2016.11.005.

[28] A. Bazzi, B. M. Masini, A. Zanella, and A. Calisti, “Visible light communications as a complementary technology for the internet of vehicles,” *Computer Communications*, vol. 93, pp. 39–51, Nov. 2016, doi: 10.1016/j.comcom.2016.07.004.

[29] Y. Lou, P. Li, and X. Hong, “A distributed framework for network-wide traffic monitoring and platoon information aggregation using V2V communications,” *Transportation Research Part C: Emerging Technologies*, vol. 69, pp. 356–374, Aug. 2016, doi: 10.1016/j.trc.2016.06.003.

[30] S. Gao, A. Lim, and D. Bevly, “An empirical study of DSRC V2V performance in truck platooning scenarios,” *Digital Communications and Networks*, vol. 2, no. 4, pp. 233–244, Nov. 2016, doi: 10.1016/j.dcan.2016.10.003.

[31] C. M. Silva, F. A. Silva, J. F. M. Sarubbi, T. R. Oliveira, W. Meira, and J. M. S. Nogueira, “Designing mobile content delivery networks for the internet of vehicles,” *Vehicular Communications*, vol. 8, pp. 45–55, Apr. 2017, doi: 10.1016/j.vehcom.2016.11.003.

[32] V. P. Harigovindan, A. V. Babu, and L. Jacob, “Proportional fair resource allocation in vehicle-to-infrastructure networks for drive-thru Internet applications,” *Computer Communications*, vol. 40, pp. 33–50, Mar. 2014, doi: 10.1016/j.comcom.2013.12.001.

[33] J. Godoy, V. Milanés, J. Pérez, J. Villagrá, and E. Onieva, “An auxiliary V2I network for road transport and dynamic environments,” *Transportation Research Part C: Emerging Technologies*, vol. 37, pp. 145–156, Dec. 2013, doi: 10.1016/j.trc.2013.09.012.

[34] J. Santa, A. F. Gómez-Skarmeta, and M. Sánchez-Artigas, “Architecture and evaluation of a unified V2V and V2I communication system based on cellular networks,” *Computer Communications*, vol. 31, no. 12, pp. 2850–2861, Jul. 2008, doi: 10.1016/j.comcom.2007.12.008.

[35] G. A. Ubiergo and W.-L. Jin, “Mobility and environment improvement of signalized networks through Vehicle-to-Infrastructure (V2I) communications,” *Transportation Research Part C: Emerging Technologies*, vol. 68, pp. 70–82, Jul. 2016, doi: 10.1016/j.trc.2016.03.010.

[36] P. Belanovic, D. Valerio, A. Paier, T. Zemen, F. Ricciato, and C. F. Mecklenbrauker, “On Wireless Links for Vehicle-to-Infrastructure Communications,” *IEEE Transactions on Vehicular Technology*, vol. 59, no. 1, pp. 269–282, Jan. 2010, doi: 10.1109/TVT.2009.2029119.

[37] R. Atallah, M. Khabbaz, and C. Assi, “Multihop V2I Communications: A Feasibility Study, Modeling, and Performance Analysis,” *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 2801–2810, Mar. 2017, doi: 10.1109/TVT.2016.2586758.

[38] F. Sakiz and S. Sen, “A survey of attacks and detection mechanisms on intelligent transportation systems: VANETs and IoV,” *Ad Hoc Networks*, vol. 61, pp. 33–50, Jun. 2017, doi: 10.1016/j.adhoc.2017.03.006.

[39] K. Sung, J. Lee, and J. Shin, “Study of CAN-to-3GPP LTE gateway architecture for automotive safety in V2I environment,” in *2015 17th International Conference on Advanced Communication Technology (ICACT)*, Jul. 2015, pp. 256–259. doi: 10.1109/ICACT.2015.7224797.

[40] T. Kopacz, A. Narbudowicz, D. Heberling, and M. J. Ammann, “Evaluation of automotive MIMO antennas for V2V communication in urban intersection scenarios,” in *2017 11th European Conference on Antennas and Propagation (EUCAP)*, Mar. 2017, pp. 2907–2911. doi: 10.23919/EuCAP.2017.7928476.

[41] H. Seo, K.-D. Lee, S. Yasukawa, Y. Peng, and P. Sartori, “LTE evolution for vehicle-to-everything services,” *IEEE Communications Magazine*, vol. 54, no. 6, pp. 22–28, Jun. 2016, doi: 10.1109/MCOM.2016.7497762.

[42] S. Fujikami, T. Sumi, R. Yagiu, and Y. Nagai, “Fast Device Discovery for Vehicle-to-Pedestrian communication using wireless LAN,” in *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*, Jan. 2015, pp. 35–40. doi: 10.1109/CCNC.2015.7157943.

[43] M. Suwa, M. Nishimura, and R. Sakata, “LED Projection Module enables a vehicle to communicate with pedestrians and other vehicles,” in *2017 IEEE International Conference on Consumer Electronics (ICCE)*, Jan. 2017, pp. 37–38. doi: 10.1109/ICCE.2017.7889220.

[44] P. Merdrignac, O. Shagdar, and F. Nashashibi, “Fusion of Perception and V2P Communication Systems for the Safety of Vulnerable Road Users,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 7, pp. 1740–1751, Jul. 2017, doi: 10.1109/TITS.2016.2627014.

[45] Y. Gao, G. G. Md. N. Ali, P. H. J. Chong, and Y. L. Guan, “Network Coding Based BSM Broadcasting at Road Intersection in V2V Communication,” in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, Sep. 2016, pp. 1–5. doi: 10.1109/VTCFall.2016.7881105.

[46] S. Temel, M. C. Vuran, and R. K. Faller, “A Primer on Vehicle-to-Barrier Communications: Effects of Roadside Barriers, Encroachment, and Vehicle Braking,” in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, Sep. 2016, pp. 1–7. doi: 10.1109/VTCFall.2016.7880871.

[47] “Integrity-oriented service scheduling for vehicle-to-roadside data access | IEEE Conference Publication | IEEE Xplore.” https://ieeexplore.ieee.org/document/7954509 (accessed Dec. 03, 2022).

[48] D. Niyato and E. Hossain, “A Unified Framework for Optimal Wireless Access for Data Streaming Over Vehicle-to-Roadside Communications,” *IEEE Transactions on Vehicular Technology*, vol. 59, no. 6, pp. 3025–3035, Jul. 2010, doi: 10.1109/TVT.2010.2048769.

[49] H. Shin and R. Baldick, “Plug-In Electric Vehicle to Home (V2H) Operation Under a Grid Outage,” *IEEE Transactions on Smart Grid*, vol. 8, no. 4, pp. 2032–2041, Jul. 2017, doi: 10.1109/TSG.2016.2603502.

[50] A. Fachechi *et al.*, “A new vehicle-to-grid system for battery charging exploiting IoT protocols,” in *2015 IEEE International Conference on Industrial Technology (ICIT)*, Mar. 2015, pp. 2154–2159. doi: 10.1109/ICIT.2015.7125414.

[51] D. Niyato, D. T. Hoang, P. Wang, and Z. Han, “Cyber Insurance for Plug-In Electric Vehicle Charging in Vehicle-to-Grid Systems,” *IEEE Network*, vol. 31, no. 2, pp. 38–46, Mar. 2017, doi: 10.1109/MNET.2017.1600321NM.

[52] S. Kumar and R. Y. U. Kumar, “Performance analysis of LTE protocol for EV to EV communication in vehicle-to-grid (V2G),” in *2015 IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE)*, May 2015, pp. 1567–1571. doi: 10.1109/CCECE.2015.7129514.

[53] R. Huang *et al.*, “Integration of IEC 61850 into a Vehicle-to-Grid system with networked electric vehicles,” in *2015 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, Feb. 2015, pp. 1–5. doi: 10.1109/ISGT.2015.7131826.

[54] S. Yoon, K. Park, and E. Hwang, “Connected electric vehicles for flexible vehicle-to-grid (V2G) services,” in *2017 International Conference on Information Networking (ICOIN)*, Jan. 2017, pp. 411–413. doi: 10.1109/ICOIN.2017.7899469.

[55] Deeksha, A. Kumar, and M. Bansal, “A review on VANET security attacks and their countermeasure,” in *2017 4th International Conference on Signal Processing, Computing and Control (ISPCC)*, Sep. 2017, pp. 580–585. doi: 10.1109/ISPCC.2017.8269745.

[56] “Fast Batch Verification of Multiple Signatures | SpringerLink.” https://link.springer.com/chapter/10.1007/978-3-540-71677-8_29 (accessed Dec. 04, 2022).

[57] E. Ben Hamida and M. A. Javed, “Channel-Aware ECDSA Signature Verification of Basic Safety Messages with K-Means Clustering in VANETs,” in *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, Mar. 2016, pp. 603–610. doi: 10.1109/AINA.2016.51.

[58] “A Cross-Layer Approach to Privacy-Preserving Authentication in WAVE-Enabled VANETs | IEEE Journals & Magazine | IEEE Xplore.” https://ieeexplore.ieee.org/document/6408238 (accessed Dec. 04, 2022).

[59] S. Biswas and J. Mišić, “Relevance-based verification of VANET safety messages,” in *2012 IEEE International Conference on Communications (ICC)*, Jun. 2012, pp. 5124–5128. doi: 10.1109/ICC.2012.6364399.

[60] J. Fadhil and Q. Sarhan, “Internet of Vehicles (IoV): A Survey of Challenges and Solutions,” Nov. 2020, pp. 1–10. doi: 10.1109/ACIT50332.2020.9300095.

[61] T. Li, C. Li, J. Luo, and L. Song, “Wireless recommendations for Internet of vehicles: Recent advances, challenges, and opportunities,” *Intell. and Converged Netw.*, vol. 1, no. 1, pp. 1–17, Jun. 2020, doi: 10.23919/ICN.2020.0005.

[62] J. Liu, S. Zhang, W. Sun, and Y. Shi, “In-Vehicle Network Attacks and Countermeasures: Challenges and Future Directions,” *IEEE Network*, vol. 31, no. 5, pp. 50–58, 2017, doi: 10.1109/MNET.2017.1600257.

[63] “Intrusion detection system based on the analysis of time intervals of CAN messages for in-vehicle network | Request PDF.” https://www.researchgate.net/publication/301258479_Intrusion_detection_system_based_on_the_analysis_of_time_intervals_of_CAN_messages_for_in-vehicle_network (accessed Dec. 04, 2022).

[64] J. Contreras-Castillo, S. Zeadally, and J. A. Guerrero-Ibañez, “Internet of Vehicles: Architecture, Protocols, and Security,” *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 3701–3709, Oct. 2018, doi: 10.1109/JIOT.2017.2690902.

[65] “(PDF) Car-2-Car Communication Consortium - Manifesto.” https://www.researchgate.net/publication/224986100_Car-2-Car_Communication_Consortium_-_Manifesto (accessed Dec. 04, 2022).

[66] “ITS Standards – Intelligent Transport Systems.” https://www.itsstandards.eu/ (accessed Dec. 04, 2022).

[67] “Safespot.” http://www.safespot-eu.org/news/cooperative_mobility_showcase_2010.html (accessed Dec. 04, 2022).

[68] N. Maslekar, M. Boussedjra, J. Mouzna, and H. Labiod, “VANET Based Adaptive Traffic Signal Control,” in *2011 IEEE 73rd Vehicular Technology Conference (VTC Spring)*, May 2011, pp. 1–5. doi: 10.1109/VETECS.2011.5956305.

[69] R. Wunderlich, C. Liu, I. Elhanany, and T. Urbanik, “A Novel Signal-Scheduling Algorithm With Quality-of-Service Provisioning for an Isolated Intersection,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 3, pp. 536–547, Sep. 2008, doi: 10.1109/TITS.2008.928266.

[70] V. Gradinescu, C. Gorgorin, R. Diaconescu, V. Cristea, and L. Iftode, “Adaptive Traffic Lights Using Car-to-Car Communication,” in *2007 IEEE 65th Vehicular Technology Conference - VTC2007-Spring*, Apr. 2007, pp. 21–25. doi: 10.1109/VETECS.2007.17.

[71] P. LA and S. Bhatnagar, “Reinforcement Learning With Function Approximation for Traffic Signal Control,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 2, pp. 412–421, Jun. 2011, doi: 10.1109/TITS.2010.2091408.

[72] “Route-Based Vehicular Traffic Management for Wireless Access in Vehicular Environments | IEEE Conference Publication | IEEE Xplore.” https://ieeexplore.ieee.org/document/4657093 (accessed Dec. 04, 2022).

[73] “VSPN: VANET-Based Secure and Privacy-Preserving Navigation | IEEE Journals & Magazine | IEEE Xplore.” https://ieeexplore.ieee.org/document/6257366 (accessed Dec. 04, 2022).

[74] P.-Y. Chen, Y.-M. Guo, and W.-T. Chen, “Fuel-Saving Navigation System in VANETs,” in *2010 IEEE 72nd Vehicular Technology Conference - Fall*, Sep. 2010, pp. 1–5. doi: 10.1109/VETECF.2010.5594424.

[75] V. Verroios, V. Efstathiou, and A. Delis, “Reaching Available Public Parking Spaces in Urban Environments Using Ad Hoc Networking,” in *2011 IEEE 12th International Conference on Mobile Data Management*, Jun. 2011, vol. 1, pp. 141–151. doi: 10.1109/MDM.2011.49.

[76] R. Lu, X. Lin, H. Zhu, and X. Shen, “SPARK: A New VANET-Based Smart Parking Scheme for Large Parking Lots,” in *IEEE INFOCOM 2009*, Apr. 2009, pp. 1413–1421. doi: 10.1109/INFCOM.2009.5062057.

[77] A. Ksentini, H. Tounsi, and M. Frikha, “A proxy-based framework for QoS-enabled Internet access in VANETS,” in *The Second International Conference on Communications and Networking*, Nov. 2010, pp. 1–8. doi: 10.1109/COMNET.2010.5699820.

[78] M. Asefi, S. Cespedes, X. Shen, and J. W. Mark, “A Seamless Quality-Driven Multi-Hop Data Delivery Scheme for Video Streaming in Urban VANET Scenarios,” in *2011 IEEE International Conference on Communications (ICC)*, Jun. 2011, pp. 1–5. doi: 10.1109/icc.2011.5962785.

[79] J. Santa, F. Pereñíguez, A. Moragón, and A. F. Skarmeta, “Experimental evaluation of CAM and DENM messaging services in vehicular communications,” *Transportation Research Part C: Emerging Technologies*, vol. 46, pp. 98–120, Sep. 2014, doi: 10.1016/j.trc.2014.05.006.

[80] “(PDF) Timing Attack in Vehicular Network.” https://www.researchgate.net/publication/260365999_Timing_Attack_in_Vehicular_Network (accessed Dec. 04, 2022).

[81] M. S. Al-kahtani, “Survey on security attacks in Vehicular Ad hoc Networks (VANETs),” in *2012 6th International Conference on Signal Processing and Communication Systems*, Dec. 2012, pp. 1–9. doi: 10.1109/ICSPCS.2012.6507953.

[82] G. Yan, S. Olariu, and M. C. Weigle, “Providing VANET security through active position detection,” *Computer Communications*, vol. 31, no. 12, pp. 2883–2897, Jul. 2008, doi: 10.1016/j.comcom.2008.01.009.

[83] “A sybil attack detection approach using neighboring vehicles in VANET | Proceedings of the 4th international conference on Security of information and networks.” https://dl.acm.org/doi/10.1145/2070425.2070450 (accessed Dec. 04, 2022).

[84] “SP 800-94, Guide to Intrusion Detection and Prevention Systems (IDPS) | CSRC.” https://csrc.nist.gov/publications/detail/sp/800-94/final (accessed Dec. 04, 2022).

[85] T. Zhang, H. Antunes, and S. Aggarwal, “Defending Connected Vehicles Against Malware: Challenges and a Solution Framework,” *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 10–21, Feb. 2014, doi: 10.1109/JIOT.2014.2302386.

[86] G. D. Putra and S. Sulistyo, “Trust Based Approach in Adjacent Vehicles to Mitigate Sybil Attacks in VANET,” in *Proceedings of the 2017 International Conference on Software and e-Business*, New York, NY, USA, Dec. 2017, pp. 117–122. doi: 10.1145/3178212.3178231.

[87] L. B. Othmane, H. Weffers, M. M. Mohamad, and M. Wolf, “A Survey of Security and Privacy in Connected Vehicles,” in *Wireless Sensor and Mobile Ad-Hoc Networks*, D. Benhaddou and A. Al-Fuqaha, Eds. New York, NY: Springer New York, 2015, pp. 217–247. doi: 10.1007/978-1-4939-2468-4_10.

[88] D. Gantsou, “On the use of security analytics for attack detection in vehicular ad hoc networks,” in *2015 International Conference on Cyber Security of Smart Cities, Industrial Control System and Communications (SSIC)*, Aug. 2015, pp. 1–6. doi: 10.1109/SSIC.2015.7245674.

[89] R. Mishra, A. Singh, and R. Kumar, “VANET security: Issues, challenges and solutions,” in *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, Mar. 2016, pp. 1050–1055. doi: 10.1109/ICEEOT.2016.7754846.

[90] H. Hasrouny, A. E. Samhat, C. Bassil, and A. Laouiti, “VANet security challenges and solutions: A survey,” *Vehicular Communications*, vol. 7, pp. 7–20, Jan. 2017, doi: 10.1016/j.vehcom.2017.01.002.

[91] S. Tbatou, A. Ramrami, and Y. Tabii, “Security of communications in connected cars Modeling and safety assessment,” in *Proceedings of the 2nd international Conference on Big Data, Cloud and Applications*, New York, NY, USA, Mar. 2017, pp. 1–7. doi: 10.1145/3090354.3090412.

[92] “Security threats in vehicular ad hoc networks | IEEE Conference Publication | IEEE Xplore.” https://ieeexplore.ieee.org/document/7732079 (accessed Dec. 04, 2022).

[93] S. S. Manvi and S. Tangade, “A survey on authentication schemes in VANETs for secured communication,” *Vehicular Communications*, vol. 9, pp. 19–30, Jul. 2017, doi: 10.1016/j.vehcom.2017.02.001.

[94] “What is Deep Learning and How Does It Work?,” *Enterprise AI*. https://www.techtarget.com/searchenterpriseai/definition/deep-learning-deep-neural-network (accessed Dec. 04, 2022).

[95] “A. Kumar and T. J. Lim, ‘EDIMA: Early Detection of IoT Malware Network Activity Using Machine Learning Techniques,’ 2019 IEEE 5th World Forum on Internet of Things (WF-IoT), Limerick, Ireland, pp. 289-294, 2019 - Google Search.” 2022).

[96] R. S. Vitalkar, S. S. Thorat, and D. V. Rojatkar, “Intrusion Detection System for Vehicular Ad-hoc Network using Deep Learning,” vol. 07, no. 12, p. 7, 2020.

[97] L. Zhang and D. Ma, “A Hybrid Approach Toward Efficient and Accurate Intrusion Detection for In-Vehicle Networks,” *IEEE Access*, vol. 10, pp. 10852–10866, 2022, doi: 10.1109/ACCESS.2022.3145007.

[98] K. Agrawal, T. Alladi, A. Agrawal, V. Chamola, and A. Benslimane, “NovelADS: A Novel Anomaly Detection System for Intra-Vehicular Networks,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, Jan. 2022, doi: 10.1109/TITS.2022.3146024.

[99] M. A. Shahid, “Fixed Cluster Based Cluster Head Selection Algorithm in Vehicular Adhoc Network,” p. 67.

[100] Iman Almomani,[1,2] Bassam Al-Kasasbeh,[2] and Mousa AL-Akhras[2,3] "WSN-DS: A Dataset for Intrusion Detection Systems in Wireless Sensor Networks"

[101] Sultan Basudan, Xiaodong Lin, *Fellow, IEEE,* and Karthik Sankaranarayanan"A Privacy-Preserving Vehicular Crowdsensing-Based Road Surface Condition Monitoring System Using Fog Computing"

Appendix 1: CH selection algorithm

The working RSU checks whether the next RSU is working or not

RSU------>check (next RSU stop working==true)

Set suitable values for W1, W2 and W3

Set D= cluster head selection interval (for how long the CH is working)

2. Determine the set (V) of vehicle in the cluster

3. Initialize parameters

- Nv=|V|
- Savg=(∑Vi ∈V Si/Nv)
- Current_best=-999
- CH=ϕ

4. For each vehicle Vi∈V

- Di=cal_dist_to_boundary(Xi,Yi)
- Fi=W1*TVi+W2*di-W3*(Si-Savg)
- If Fi>current_best && Fi+1>current_best

I. Current_best= Fi && current_best=Fi+1

II. Main CH=Vi and reserved CH=Vi+1

III. Push the deep learning classifier to Vi && Vi+1

Appendix 2: deep neural network model code taken from kaggle

```
import pandas aspd
importnumpyas np

importmatplotlib.pyplotasplt
frommatplotlib.pyplotimport figure
importseabornassns

fromsklearn.metricsimportconfusion_matrix
fromsklearn.metricsimportaccuracy_score
fromsklearn.metricsimportclassification_report
fromsklearn.model_selectionimporttrain_test_split
fromsklearnimport metrics
fromsklearn.model_selectionimportcross_val_score
fromsklearnimport preprocessing

import time


fromsklearnimport metrics
importos
fordirname, _, filenames inos.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
data = pd.read_csv('/kaggle/input/wsnds/WSN-
DS.csv',header=0,names=['id','time','Is_CH','who_CH','Dist_To_CH','ADV_S','ADV_R','JOIN_S
','JOIN_R','SCH_S','SCH_R','Rank','DATA_S','DATA_R','Data_Sent_To_BS','dist_CH_To_BS','
send_code','Expaned_Energy','label'])
data=data.drop(['Data_Sent_To_BS','dist_CH_To_BS','Expaned_Energy'],axis=1)
data['label'] = data['label'].replace(['Normal'], 0)
data['label'] = data['label'].replace(['Flooding'], 1)
data['label'] = data['label'].replace(['TDMA'], 2)
data['label'] = data['label'].replace(['Grayhole'], 3)
data['label'] = data['label'].replace(['Blackhole'], 4)
rescale = data[data['label'] >0]
#rescale = pd.DataFrame(np.repeat(rescale.to_numpy(), 6,axis=0),columns=rescale.columns)
label_dict = dict(data.label.value_counts())
sns.countplot(data.label)
labels = ["Normal",'Flooding','TDMA','Grayhole','Blackhole']
sizes = [dict(data.label.value_counts())[0], dict(data.label.value_counts())[1],
dict(data.label.value_counts())[2], dict(data.label.value_counts())[3],
dict(data.label.value_counts())[4]]
plt.figure(figsize = (13,8))
plt.pie(sizes, labels=labels, autopct='%1.1f%%',
        shadow=True, startangle=90)
plt.legend( ["Normal",'Flooding','TDMA','Grayhole','Blackhole'])
plt.title('The percentage of Normal and Maliciuos Requests in dataset')
plt.show()
figure(figsize=(9, 5), dpi=80)
data[data.columns[data.isna().sum() >= 0]].isna().sum().sort_values().plot.bar()
plt.title("Features which has NuLL values")
figure(figsize=(12, 7), dpi=80)
plt.barh(list(dict(data.SCH_R.value_counts()).keys()),
dict(data.SCH_R.value_counts()).values(), color='lawngreen')
```

```
plt.barh(list(dict(data[data.label == 1].SCH_R.value_counts()).keys()), dict(data[data.label ==
1].SCH_R.value_counts()).values(), color='blue')

foridx, valinenumerate(dict(data.SCH_R.value_counts()).values()):
    plt.text(x = val, y = idx-0.2, s = str(val), color='r', size = 13)

foridx, valinenumerate(dict(data[data.label == 1].SCH_R.value_counts()).values()):
    plt.text(x = val, y = idx-0.2, s = str(val), color='w', size = 13)


plt.xlabel('Number of Requests')
plt.ylabel('IP addres of sender')
plt.legend(['All','malicious'])
plt.title('Number of requests from different IP adress')
fromimblearn.over_sampling import SMOTE
from collections import Counter
frommatplotlibimportpyplot
oversample = SMOTE()
X, y = oversample.fit_resample(xx, yy)
dataset = pd.DataFrame(X, columns =
['id','time','Is_CH','who_CH','Dist_To_CH','ADV_S','ADV_R','JOIN_S','JOIN_R',
                                          'SCH_S','SCH_R','Rank','DATA_S','DATA_R',
                                          'send_code','label'])

unbalanced = pd.DataFrame(xx, columns =
['id','time','Is_CH','who_CH','Dist_To_CH','ADV_S','ADV_R','JOIN_S','JOIN_R',
                                          'SCH_S','SCH_R','Rank','DATA_S','DATA_R',
                                          'send_code','label'])
label_dict = dict(unbalanced.label.value_counts())
sns.countplot(unbalanced.label)
label_dict = dict(dataset.label.value_counts())
sns.countplot(dataset.label)
train,test=train_test_split(dataset,test_size=0.3, random_state=45,shuffle=True)
fromsklearn.preprocessingimportMinMaxScaler
min_max_scaler =
MinMaxScaler().fit(train[['time','Is_CH','who_CH','Dist_To_CH','ADV_S','ADV_R','JOIN_S','J
OIN_R',
                                          'SCH_S','SCH_R','Rank','DATA_S','DATA_R',
                                          'send_code']])
numerical_columns=['time','Is_CH','who_CH','Dist_To_CH','ADV_S','ADV_R','JOIN_S','JOIN_
R',
                                          'SCH_S','SCH_R','Rank','DATA_S','DATA_R',
                                          'send_code']
train[numerical_columns] = min_max_scaler.transform(train[numerical_columns])
print("Full dataset:\n")
print("Normal: " + str(data["label"].value_counts()[[0]].sum()))
print("Flooding: " + str(data["label"].value_counts()[[1]].sum()))
print("TDMA: " + str(data["label"].value_counts()[[2]].sum()))
print("Grayhole: " + str(data["label"].value_counts()[[3]].sum()))
print("Blackhole: " + str(data["label"].value_counts()[[4]].sum()))
print("---------------")

print("Training set:\n")
print("Normal: " + str(train["label"].value_counts()[[0]].sum()))
print("Flooding: " + str(train["label"].value_counts()[[1]].sum()))
print("TDMA: " + str(train["label"].value_counts()[[2]].sum()))
print("Grayhole: " + str(train["label"].value_counts()[[3]].sum()))
print("Blackhole: " + str(train["label"].value_counts()[[4]].sum()))
print("---------------")

print("Test set:\n")
print("Normal: " + str(test["label"].value_counts()[[0]].sum()))
print("Flooding: " + str(test["label"].value_counts()[[1]].sum()))
print("TDMA: " + str(test["label"].value_counts()[[2]].sum()))
print("Grayhole: " + str(test["label"].value_counts()[[3]].sum()))
print("Blackhole: " + str(test["label"].value_counts()[[4]].sum()))
X_train = train.values
print(type(X_train))
print(type(y_train))
print(X_train.shape)
print(y_train.shape)
y_test=test.copy()
y_test = np.array(y_test.pop("label"))

X_test = test.values

print(type(X_test))
print(type(y_test))
print(X_test.shape)
print(y_test.shape)
x_val = X_train[-100000:]
y_val = y_train[-100000:]
X_train = X_train[:-100000]
y_train = y_train[:-100000]
importtensorflowastf
fromtensorflowimportkeras
fromtensorflow.kerasimport layers
fromkeras.layersimport Input,
Dense,BatchNormalization,Dropout,LSTM,Concatenate,Bidirectional,Flatten,InputLayer,Minim
um
```

```
fromkeras.modelsimportModel,Sequential


MLP_X_train=X_train.copy()
LSTM_X_train=X_train.copy()

MLP_X_test=X_test.copy()
LSTM_X_test=X_test.copy()

LSTM_X_val=x_val.copy()
LSTM_X_train = np.reshape(LSTM_X_train, (LSTM_X_train.shape[0], 16,1 ))
LSTM_X_test = np.reshape(LSTM_X_test, (LSTM_X_test.shape[0], 16,1))
LSTM_x_val= np.reshape(LSTM_X_val, (LSTM_X_val.shape[0], 16,1))
LSTM_X_test.shape
importkeras.backendas K
K.clear_session()
#LSTM Model
#kernel_regularizer=keras.regularizers.l2(0.001),
input1= keras.Input(shape=LSTM_X_train.shape[1:])
lstm1 = LSTM(200)(input1)
LSTM_drp=Dropout(rate = 0.2)(lstm1)
lstm_dns=Dense(75,kernel_regularizer=keras.regularizers.l2(0.001),
activation='relu')(LSTM_drp)
lstm_drp1=Dropout(rate = 0.3)(lstm_dns)
lstm_dns1=Dense(50,activation='relu')(lstm_drp1)
lstm=Dropout(rate = 0.2)(lstm_dns1)

#MLP Model
input2= keras.Input(shape=MLP_X_train.shape[1] )
ann=Dense(200,kernel_regularizer=keras.regularizers.l2(0.001),activation='relu')(input2)
ann_bn=BatchNormalization()(ann)
MLP_drp=Dropout(rate = 0.2)(ann_bn)
dns=Dense(75, activation='relu')(MLP_drp)
drp1=Dropout(rate = 0.2)(dns)
dns1=Dense(50, activation='relu')(drp1)
MLP=Dropout(rate = 0.2)(dns1)


#Merging The two layers
merged=Concatenate()([MLP,lstm])
bnorm2=BatchNormalization()(merged)
ndss=Dense(50,kernel_regularizer=keras.regularizers.l2(0.001), activation='relu')(bnorm2)
ndss1=Dense(25, activation='relu')(ndss)
ndss2=Dense(50, activation='relu')(ndss1)
ndss3=Dense(25, activation='relu')(ndss2)
```

```
merg_drp=Dropout(rate = 0.2)(ndss3)
output=Dense(5,activation='softmax')(merg_drp)
model = Model(inputs=[(input1, input2)], outputs=output)

model.summary()
K.clear_session()
fromtensorflow.kerasimport callbacks
early_stopping = callbacks.EarlyStopping(
    min_delta = 0.01,
    monitor="val_loss",
    patience = 4,
    #restore_best_weights = True
)
model.compile(
    optimizer = keras.optimizers.Adam(learning_rate=0.0001),
    loss=keras.losses.SparseCategoricalCrossentropy(),
    metrics=[keras.metrics.SparseCategoricalAccuracy()]
)
history1 = model.fit(
    [(LSTM_X_train,MLP_X_train)], y_train,
    epochs=10,
    verbose =1,
    validation_data=([LSTM_X_val,x_val], y_val),
    batch_size = 256,
    callbacks=[early_stopping]
)
#validation_data=(x_val, y_val),
#validation_data=([LSTM_X_val,x_val], y_val)

frommatplotlibimportpyplotasplt

plt.plot(history1.history['sparse_categorical_accuracy'])
plt.plot(history1.history['val_sparse_categorical_accuracy'])
plt.title('model accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train', 'val'], loc='upper left')
plt.show()
history_frame = pd.DataFrame(history1.history)
history_frame.loc[:, ['loss']].plot()
history_frame.loc[:, ['sparse_categorical_accuracy']].plot();
fromsklearn.metricsimportconfusion_matrix
y_pred = model.predict([LSTM_X_test,MLP_X_test])
ll = np.argmax(y_pred, axis=1)
```

```
confusion = tf.math.confusion_matrix(
          labels = y_test,      # get true labels
          predictions = ll,  # get predicted labels
          num_classes=5)             # no. of classifier
#confusion = confusion_matrix(y_test, y_pred)
print('Confusion Matrix\n')
print(confusion)
importseabornassns
import pandas aspd

cm = pd.DataFrame(confusion.numpy(), # use .numpy(), because now confusion is tensor
           range(5),range(5))

cm_df = pd.DataFrame(cm,
               index = ['Normal','Flooding','TDMA','Grayhole','Blackhole'],
               columns = ['Normal','Flooding','TDMA','Grayhole','Blackhole'])
plt.figure(figsize = (10,10))
sns.heatmap(cm, annot=True, annot_kws={"size": 12}) # font size
plt.ylabel('Actal Values')
plt.xlabel('Predicted Values')
plt.show()
fromsklearn.metricsimportclassification_report
report = classification_report(y_test, ll, labels=[0,1,2,3,4], target_names=["Normal", "Flooding",
"TDMA","Grayhole","Blackhole"])
print(report)
fromsklearn.metricsimportconfusion_matrix
cf_matrix = confusion_matrix(y_test, ll)
importseabornassns
sns.heatmap(cf_matrix, annot=True)
importdatetimeasdt
fromsklearn.metricsimportaccuracy_score, confusion_matrix, precision_score, recall_score,
f1_score, roc_auc_score

# Measure accuracy
print('Predicting on the test data:')
start = dt.datetime.now()
escore = model.evaluate([LSTM_X_test,MLP_X_test], y_test, batch_size=128)
#pred = model.predict([LSTM_X_test,MLP_X_test])
#pred = np.argmax(pred)
```