INTELLIGENT ROAD ACCIDENT DETECTION IN IOVs

J COMPONENT PROJECT REPORT
SLOT – A1

Submitted by

Harsha Lokesh 19BCT0240 harsha.lokesh2019@vitstudent.ac.in

Rahul Bhassy 19BCT0099 rahul.bhassy2019@vitstudent.ac.in

Nikhil Janga 19BCT0244 janganikhil.jospeh2019@vitstudent.ac.in

Joel Daniel Johnson 19BCT0263 joeldanie.johnson2019@vitstudent.ac.in

Course Code: BCT3001
Course Title: Wireless Ad-hoc and Sensor Networks
Under the guidance of
S. Ananda Kumar
B. Tech
in
Computer Science and Engineering with specialization in IoT.

School of Computer Science and Engineering

Dec, 2021

Table of content

S. No	Contents	Page number
1	Abstract	3
2	Introduction	3
3	Flow diagram / Literature review	4
4	Proposed solution	32
5	Data visualization and model implementation	33
6	Accuracy table	43
7	Conclusion/ future works	44
8	Survey table	45
9	References	46
10	Dataset/video links	48

Intelligent road accident detection in IoV (Internet of Vehicles)

Harsha Lokesh -19BCT0240
Rahul Bhassy – 19BCT0099
Nikhil Janga -19BCT0244
Joel Daniel Johnson -19BCT0263

VIT – Vellore Institute of Technology, Vellore

Abstract: -

Road accidents are one of the highest causes of death and injuries world over, which can not only be prevented but also avoided all together, modern technological advancements with the use of interconnection of Vehicles and WSN. In this paper we provide a comprehensive study and summary of papers needed for road accident detection, smart alert systems and accident avoidance, by making use of intelligent routing and providing machine learning models for adaptive internet of vehicles. The machine learning model provide an accurate model for accident detection of vehicles, which can be predicted with the sensor data and predefined models.

Keyword: -

Wireless sensor network, V2X communication, VANET, Machine learning algorithms, routing.

INTRODUCTION

In this modern world transportation plays an important role in our lives. Transportation of people, goods and much more became more efficient and faster with advancement in transport technology. Due to advances in transportation technology the number of vehicles is also drastically increasing. However, with the rising number of vehicles, the accident cases are also rising. Estimates show that more than a million people die every year as a result of vehicle accidents. There are many risk factors associated with accidents, they are over speeding, driving under influence of alcohol and other drugs, distracted driving, lack of proper safety equipment's like seatbelts, helmets, unsafe road and vehicles, inadequate post-crash care, lack of proper traffic enforcement. More than 90% traffic accidents happen in developing and under developed countries. The leading cause of death for young adults are traffic accidents. Majority of fatalities happen due to lack of post accidental health care. This case is especially high in low- and middle-income countries. And many non-fatal accident victims suffer various disabilities throughout their life. In 2017 WHO released a safety technical package called save LIVES which focuses on Speed management, Leadership, infrastructure design and improvement, vehicle safety standards. With the advances in technologies like IOT it is possible to prevent accidents by an extend and help accident victims to receive post-accident emergency healthcare which can potentially save lives. The services that IOT provide is a boon to the healthcare systems. Currently vehicle manufacturers are using IoT applications in the vehicles to increase the safety of passengers [28]. Many vehicles are in built with automatic crash response system that can communicate with nearest medical care unit using cloud services [28]. Furthermore, for accident prevention sensors are able to detect over-speeding vehicles, alcohol content which can prevent accidents from happening. For accident detection Biomedical sensors such as pulse sensors, muscle sensors are used a sudden change in these vital readings gives an emergency alert to nearest medical center. These accident alert systems help for a quick post-accident healthcare and can also alert other drivers in the area about the accident through VANETs. An IoV system for high-speed head-on and single-vehicle accident detection, analysis, and emergency notification is composed of sensors, actuators, heterogeneous networks, rescue service centers, and a cloud-based management platform. The sensors and actuators inbuilt in the vehicle sense and take actions in an accident scenario. A cloud service managed platform receives the emergency alerts and transmits the messages to nearby medical centers and ambulances.

In this paper we aim to work on an IOV system that can warn the drivers about the approaching accident-prone areas by using machine learning models. We also aim to show the build and workings of an automatic emergency alert system in the event of an accident using machine learning model. Using the various sensors, we will be able to monitor the vitals of an accident victim potentially speeding up the emergency health care.

FLOW DIAGRAM

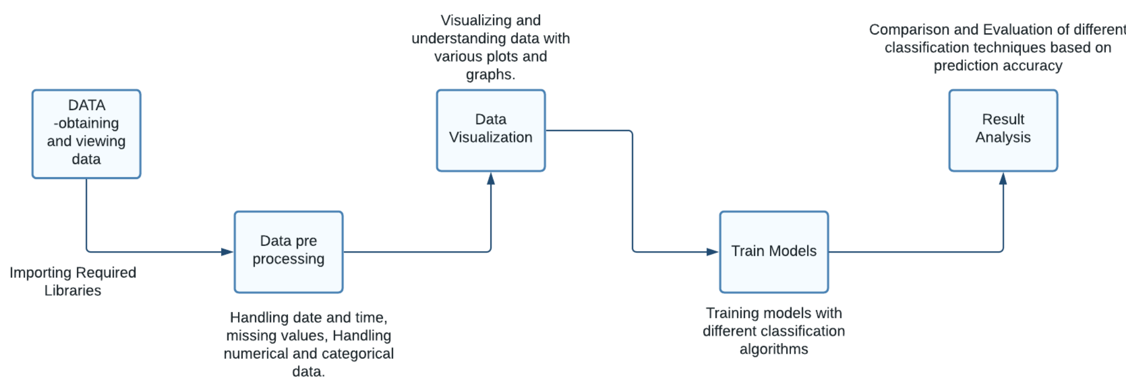


Fig 1. model flow process

NIKHIL JANGA 19BCT0244

[1] S. Bhuvaneswari and D. R. Saranya, "Internet of Vehicle based Accident Detection and Management Techniques by using VANET: An Empirical Study," *IEEE*, pp. 237-244, 2020.

This paper offers a review of the various IoV-based accident detection and management strategies that are currently in use with VANET. It also assesses current smart helmet management best practices and challenges in terms of software and hardware.

To detect alcohol and prevent accidents in two-wheelers, the first technique is to employ a smarter monitoring system with a ZigBee and RF communication module. The MQ3 alcohol sensor is used in conjunction with the ADXL335 accelerometer to detect alcohol with a rapid reaction time and a LCD display in the helmet unit, and the passive flex sensors are connected to the Arduino USB board in the helmet unit, which certifies helmet usage. The RF receiver, the ignition system and the power supply in the bike unit has been used to link all of the components, including the relay. The versatile GPS tracker with GSM module has been used for communication as well as location tracking, and a SIM card must be inserted into the GSM module to activate the communication, which sends messages to the user's family as well as the ambulance in the event of a collision, similar to how mobile phones work. This technology has a number of flaws, including the inability to identify accidents if the helmet or other systems such as GSM and GPS have been damaged as a result of an accident, as well as the inability to detect accidents if the power supply has been disturbed.

An intelligent helmet for bikers that used GSM and GPS and had two phases: transmitter and receiver was another technique. In the transmitter phase, the ATmega8 microcontroller was combined with the high temperature Limit Switch (to ensure that the helmet is properly worn), accelerometer, power supply, and HC 05 Bluetooth module. In the receiver phase, the ATmega-16 was utilized to connect the HC 06 Bluetooth module, relay, SIM 808 GPS, and GSM modem. The Bluetooth modules in the transducer are used to disseminate information. The transmitter (HC05 Bluetooth module) and receiver (HC06 Bluetooth module) Bluetooth modules will interact. The accelerometer is utilized to detect an accident in this situation because of its reliance on a previously determined threshold value. The problem of this method is that the limit switch in the helmet must always be pressed, which could cause the rider to become distracted while driving.

The intuitive ease and use of data is heightened and enhanced as a result of rapid technological advancements. Finally, the primary intelligent vehicle detection interpretations and technologies of the current state of smart helmet systems are addressed in our comprehensive review. Finally, the comprehensive examination of hardware and software features based on SWOT analysis effectively elevated the foundation of our main design factors. In the future, we intend to do additional research that focuses on cloud cyber-attack security in the IoT ecosystem.

NIKHIL JANGA 19BCT0244

[2] S. H. Hadi, A. Saha, F. Ahmad, M. S. Hasan and M. H. Milon, "A Smart Accident Detection and Control System in Vehicular Networks," *IEEE,* pp. 1-6, 2018.

A on a fog cloud-based IoV variation of mobile cloud computing, in which the Internet and vehicular cloud can effectively collaborate in IoV. Furthermore, due to the increasing reliance on wireless communication and computer technologies, IoV is increasingly vulnerable to a variety of threats. The primary goal of this is to identify the protocol of AKM-IoV significant security weaknesses. An attacker can easily launch vehicle and fog impersonation attacks against AKM-IoV, Furthermore, their suggested protocol is vulnerable to impersonation assaults on roadside units and cloud servers.

The protocol is vulnerable to automobile impersonation, fog server impersonation, road side unit impersonation, and cloud server impersonation assaults, according to cryptanalysis. All of these impersonation attacks are achievable utilizing the stolen verifier attack, which is a frequent and realistic attack in authentication schemes Although Wazid et al. claim that their scheme is secure against impersonation attacks, they also mention in section V-B of their own article that if an adversary attempts to generate valid authentication messages on behalf of any legitimate entity, he must have knowledge of that entity's secret credentials of the entity.

In a fog computing-based Internet of Vehicles, the network model illustrates communication among many players (IoV). Various sorts of entities are included in this network model, including trusted authority (TA), automobiles, fog servers, cloud servers, and roadside units (RSUs). Vehicle to vehicle (V2V), vehicle to fog server (V2F), vehicle to RSU (V2R), fog server to cloud server (F2C), and RSU to fog server (R2F) are some of the communication modes available. The (TA) is in charge of registering all of the entities before they are deployed in the network. After all of the credentials have been loaded into memory, RSUs are deployed. The OBU retains information about each vehicle, cloud server, and fog server in order to use it in subsequent authentication key management processes. Every car has an OBU, which stores and processes vehicle information. Furthermore, each car is connected to the Internet, allowing it to receive and transmit critical information. The Internet is used to communicate between R2F, V2F, V2V, V2R, and F2C.

NIKHIL JANGA 19BCT0244

[3] S. H. Hadi, A. Saha, F. Ahmad, M. S. Hasan and M. H. Milon, "A Smart Accident Detection and Control System in Vehicular Networks," *IEEE,* pp. 1-6, 2018.

A smart accident detection and control system (SAD-CS) is proposed in this research. When an accident occurs in this system, SAD-CS units automatically identify the accident by receiving the emergency code from the affected vehicles and taking the required procedures to assist prevent subsequent accidents.

When a vehicle reaches the SAD-CS zone, it tunes in to the wireless radio frequency. The microcontroller is responsible for making connections between the vehicles and the Access Point

Infrastructure (API). To establish effective communications between the vehicles and the API, the data transfer system employs the push and pull approaches. For broadcasting messages to the vehicles, the infrastructure uses the push approach. Vehicles, on the other hand, use the pull approach to retrieve data from the infrastructure. Vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and infrastructure-to-vehicle (I2V) communications can all be used depending on the needs of the circumstance.

When a vehicle enters the SAD-CS area, it is required to go through the RFID registration process. The number of vehicles on the road is denoted by Vn. The API receives the vehicle's unique RFID number. The RFID number is then sent to the CR via CBI via API. All of the vehicle's RFID numbers are stored by CR. The approximate number of vehicles that have entered the lane can be computed. When an accident happens, the SAD-CS message transmission process happens. According to the SAD-CS system, the RFID-A tag is obtained by the Access Point Infrastructure (API) from the impacted vehicles (V1 and V2), where the A in RFID is variable. It features predefined values for several sensors, such as a gas sensor for gas leaks, a fire sensor for a fire, and a shock sensor for a huge push, to identify the affected scenarios. After receiving RFID-A, API combines it with an emergency code to create RFID-AE, which is then sent to Control Room (CR) via Control Box Infrastructure (CBI). SAD-CS is capable of not only detecting the accident and sending an emergency message to other vehicles to prevent more accidents, but also of taking the required procedures to assist the affected persons and clear the impacted area. The Control Room (CR) can easily notify the nearest hospital, fire station, etc. about the location of the impacted region. As a result, timely emergency services can be provided in the impacted region.

The satellite and airbag systems can be used to update the system for long-distance accident detection. Eye blink sensors can be used to track the driver's blink rate in order to detect tiredness. For improved precision and efficiency, high-tech sensors can be used. Virtual simulations will be used to demonstrate performance evaluation.

The system relied on Raspberry Pi and Microcontrollers, with each vehicle requiring an on-board device. While the system is capable of detecting the position of the afflicted vehicles and simply measuring the appropriate steps to get rid of the worst scenarios, it is lifesaving in favor of accident detections during the occurrences instantly. It is a digital technology that automatically detects the unavoidable circumstances on the road. If every vehicle on the road is equipped with modern technology, the suggested SAD-CS can outperform the current manual traffic control management system in terms of reducing road accidents.

NIKHIL JANGA 19BCT0244

[4] Y. Kang, H. Yin and C. Berger, "Test Your Self-Driving Algorithm: An Overview of Publicly Available Driving Datasets and Virtual Testing Environments," *IEEE*, vol. 4, pp. 171-185, 2019.

The purpose of the research paper was to carefully perform a broad survey in order to present a complete overview of existing datasets and virtual testing environments that enable autonomous driving and algorithm research and development. This research focuses on datasets and virtual testing environments that may be found through organized online searches and methodical snowballing and are available to academics and developers. We focused primarily on ground truth driving data collected on public roads with partial or full open access for the datasets. A comprehensive analysis per data set or virtual testing environment is extremely customized to certain use-cases of autonomous driving development or evaluation. The usual method for testing algorithms is to use data from the target platform that has been recorded. While recorded data has the highest level of fidelity in terms of accuracy, in terms of realism, it can only be utilized for open-loop testing, such as stimulating perception algorithms. Furthermore, with recent advancements in machine learning requiring enormous amounts of data for training purposes for end-to-end learning, such approaches have gotten a lot of attention. However, systematic restrictions such as only being suitable for open-loop testing and practical limitations such as time, weather, and vehicular recording platforms limit the importance of such data collection systems in practice. To address the expanding testing needs for autonomous driving systems on SAE Level 4 and 5, complementary approaches that overcome the limitations of open-loop use on the one hand while allowing for scalability on the other are required. In terms of virtual test environments for traffic flow simulation, the open-source SUMO attracts a lot of interest due to its microscopic and multimodal traffic properties, while PTV Vissum is also competitive in integrating active traffic management and the geometry of Highway and the design of intersections. From an ADAS controller application point of view, ASM Traffic (dSpace) is typically designed for HIL testing of electronic control units (ECUs) or for early function validation by offline simulation.

In terms of virtual testing environments for traffic flow simulation, the open-source SUMO is popular due to its microscopic, multi-modal traffic characteristics, while PTV Vissum is also competitive due to its integration of active traffic management and road and intersection design geometry. ASM Traffic (dSpace) is typically designed for HIL testing of electronic control units (ECUs) or for early function validation by offline simulation from the perspective of an ADAS controller application. The global race to develop, evaluate, and deploy algorithms and solutions to realize self-driving vehicles has significantly heated up – first solutions at SAE Level 3 are being made available to customers addressing automated driving on highways for example. The community around this comprises researchers, major automotive OEMs, as well as young start-ups. They all have in common that they need open- loop and closed-loop solutions to systematically test and evaluate their approaches. The bar for contributing to the field is high, especially for researchers and young start-ups, because collecting original datasets is resource- and time-intensive.

NIKHIL JANGA 19BCT0244

[5] L. Alouache, N. Nguyen, M. Aliouat and R. Chelouah, "Survey on IoV routing protocols: Security and network architecture," *WILEY*, vol. 32, no. 2, 2018.

Routing protocols' major goal is to ensure the dependability, resilience, and quality of communications in vehicular networks, because these connections are intermittent due to vehicle mobility, speed, or the existence of barriers. Data integrity, as well as the availability of data and services, must be guaranteed through routing protocols. Different taxonomies could be used to classify them, such as the transmission strategy and the required information to construct routes, delay sensitivity, scenario dimension, and network type. When building routing protocols, security is a crucial consideration. Different security mechanisms, including as key management, intrusion detection systems, trust management, and secure routing, can be used to enable the securitization of data and protocols. In addition, new paradigms such as SDN should be considered while designing network architecture for IoV communication systems to set intelligence, reactivity, and adaptability. The control plane, which can be centralized, decentralized, or mixed, can be used to find the best data channel for the vehicle SDN architecture.

A routing protocol design should follow a communication strategy that is (1) adaptive in order to be adjusted to the requirements of the application users and to circumvent the constraints of communication in order to meet the requirements of the application users and to circumvent the constraints of communication. Reactive to circumvent black zones and assure error recovery, and finally secured to maintain the confidentiality and integrity of data, therefore the anonymity of users. In addition to improving communication performance, SDN architecture enhances security.

The SDN architecture is also used for platooning management services, with the goal of keeping the convoy together for as long as possible, particularly during the merge and split maneuvers. As a result, during these movements, a lot of information is disseminated both inside and outside the convoy, and this information is vulnerable to jamming, replay, and other security assaults. As a result, in order to deal with the numerous attacks that the convoy may be subjected to, we require a global vision, such as a global clock, for SDN applications. Road safety, traffic management, and user comfort are the three areas in which these applications and services fall. Among the possible uses in the first category are emergency and surveillance services, signaling and collision alerts, and so forth. The most representative services in the second group are traffic signal and platooning management, GPS navigation, and smart parking. Internet services such as content downloading, cloud access, on-board gaming, and commercial and marketing services fall under the third group. Furthermore, SDN controllers can activate an anchor zone to send data to vehicles that are currently far away but approaching the parking, and the vehicles can then share the data via wireless V2V communication. Furthermore, because this service is built on geocast, it is extremely sensitive to attacks that block data from being received informing about available locations. To deal with the difficulty of jamming transmission, the SDN's intelligence intervenes.

NIKHIL JANGA 19BCT0244

[6] N. M. Adelsamee, S. S. Alsaleh and A. Algarni, "On Simulating Internet of Vehicles," *IEEE*, pp. 1-9, 2018.

This research paper is based on simulating IoV. The simulation environment is split into two sections: road traffic simulation and networking simulation. The road traffic module can be a realistic or fictitious model that reflects street layouts, vehicular movements, and traffic management procedures. The networking simulation module, on the other hand, is used to mimic vehicle communication. Before beginning the simulation, it is necessary to comprehend the mobility model, which depicts the real-world behavior of vehicular traffic. As a result, we must deal with vehicle mobility, distinguish between macro- and micro-mobility descriptions. Modeling of mobility Road structure (unidirectional or bidirectional, single- or multi-lane), road features (speed limits, vehicle-class-based restrictions), and the presence of traffic signs (stop signs, traffic lights, etc.) are all controlled by macro-mobility. Micro-mobility, on the other hand, encompasses all aspects of an individual car's speed and acceleration modelling, which is responsible for speed variations, car queues, traffic jams, and overtaking. There are two types of road traffic simulations: real mobility and synthetic mobility. The road topology is extracted from real maps such as Tiger map or Open Street Map in the Real Mobility simulations. The road topology is manually entered in the Synthetic mobility simulation by listing the vertices of the graph and their interconnecting edges.

A powerful simulation tool for assessing network designs and protocols is required. However, in order to illustrate real-world systems, an exact mobility model is required. For the purpose of testing VANET, a realistic mobility model has been evaluated. When compared to alternative models such as Random Waypoint, the proposed mobility model, which is compatible with NS-2, has proven to produce more realistic results.

Open Street Map is used to upload the map to the SUMO. SUMO is then utilised to create a road traffic map. The road traffic map is then sent to Plexe, a networking simulation programme that simulates wireless communication between platoon vehicles.

Plexeis is a Veins-based system that combines the OMNeT++ network simulator with the SUMO traffic simulator. In addition, the R programming language,is used which is a sophisticated statistical framework for parsing, processing, and plotting data produced from OMNeT++ simulations. It includes a number of cars as well as base stations along the highway. SUMO was used to design the road and track the cars' movements. Then, using OMNET++, we investigated data sharing across network cars. We used Ad hoc On-demand Distance Vector Routing (AODV) as a routing protocol to create a VANET data transmission system. Furthermore, utilising the SUMO tool in OMNET ++, an actual model was employed to simulate vehicle movement in vehicular-based ad hoc network applications.

JOEL DANIEL JOHNSON 19BCT0263

[7] U. Alvi, M. A. K. Khattak, B. Shabir, A. W. Malik and S. R. Muhammad, "A Comprehensive Study on IoT Based Accident Detection Systems for Smart Vehicles," *IEEE*, vol. 8, pp. 122480-122497, 2020.

Every vehicle can be considered as a node where VANET is incorporated. And they can communicate temporarily with other vehicles nearby. And the probability of accident is calculated from the inputs received by all the nodes together.

Or by using pressure and tilt sensors. Where the outward force experienced by a vehicle is monitored with the help of an external pressure sensor to detect the occurrence of an accident. Shock sensors and GPS- detects and informs the public safety organization of the occurrence of an incident so they can take steps to minimize the loss incurred.

Using GPS and GSM- the GPS will monitor the speed of the vehicle constantly and when it drops beyond a present threshold limit it can detect the situation as an accident and in this way using GSM notify the necessary people about the incident.

Using AI/ML detection techniques- via crash prediction models and more to predict the event of an incident and in the process informing the driver to take caution.

Using Drunk driving detection techniques- is used to identify the alertness and sleepiness of the driver by placing cameras to capture the face of the driver and detect the state of his body and mind. This not only detects an accident but is also capable of stopping an accident.

Using smart helmet. - where the shock sensors in the helmet can detect if an incident has occurred or not

JOEL DANIEL JOHNSON 19BCT0263

[8] G. Bresson, Z. Alsayed, L. Yu and S. Glaser, "Simultaneous Localization and Mapping: A Survey of Current Trends in Autonomous Driving," *IEEE*, vol. 2, pp. 194-220, 2017.

Six criteria are required to be satisfied, in order to use SLAM in autonomous vehicles. Accuracy, Scalability, Availability, Recovering, Updatability and Dynamicity, while estimating a vehicle's pose, i.e., its location and orientation we have to build a model of the environment the vehicle is present in. There are two groups- Filtering where state is estimated and then updated on the go and smoothing where full trajectories are estimated using a complete set of measurements.

Pose Graph Optimization- tracks the odometry while mapping the environment to create a perfect map. But in actual cases. It's impossible to get perfect lidar mapping or odometry. This is where SLAM becomes helpful. SLAM using Pose Graph Optimization gives us more accurate mapping.

Single vehicle SLAM, multivehicle SLAM- centralized SLAM, centralized offline SLAM, decentralized SLAM.

Localization algorithms need to be improved greatly.

Algorithm

Step 1- Lidar scans the environment

Step 2- The vehicle moves, and the odometry records the coordinates of motion.

Step 3- Lidar scans the new environment and builds it into a pose graph

Step 4- After the scanning of environment is complete. It corrects any error in odometry by pose graph optimization.

Step 5- Mapping is complete after the calibration of the map

JOEL DANIEL JOHNSON 19BCT0263

[9] M. Brossard, A. Barrau and S. Bonnabel, "AI-IMU Dead-Reckoning," *arXiv*, 2019.

Autonomous vehicles need to self-localize by themselves utilizing their sensors like odometers, IMUs, radars or LiDARs, and cameras. This proposed approach uses only IMU (Inertial Measurement Unit) to find its present coordinates. Dead Reckoning is the process of finding its present location with respect to a point where it used to be in the past with the help of determined positions, the angle between them, etc.

Training of IMU:

Step 1: sample a portion of the given dataset.

ii) procure the filter estimates, compute losses and gradients with respect to its learnable parameters.

ii) update the parameters with the optimized gradient

JOEL DANIEL JOHNSON 19BCT0263

[10] J. Cheng, J. Cheng, M. Zhou, F. Liu, S. Gao and C. Liu, "Routing in Internet of Vehicles: A Review," *IEEE,* vol. 5, no. 2339-2352, p. 16, 2015.

The routing taxonomy can be defined from 5 different perspectives. The transmission strategy, the information used, delay sensitivity, dimensions of scenarios and network architectures. Unicast Routing protocols- the major goal is to transmit from a single source to a single destination. Geocast Routing protocols- it delivers a packet from its source to other nodes based on its geographical location.

Broadcast Routing Protocols: Broadcast is a routing method used to transmit traffic, weather, emergency road status, et al.

Traffic Simulation.

Vehicular mobility models are divided into 4 classes: synthetics mathematical models, survey-based ones that use mobility patterns from surveys, trace-based ones that generate mobility patterns from the survey. Trace based ones that utilize mobility patterns from actual mobility traces. And finally, traffic simulator where mobility extracted from sophisticated and realistic traffic simulators, based on data they require, we have four categories: topology, position, outline and path-based ones. Most classical ones from MANET drop into the primary category.

Since of the energetic nature of the versatile hubs in IoV, finding and keeping up courses is exceptionally challenging with them. Subsequently, re- searchers propose other sorts of steering conventions. Most of them point to unafflicted result.

Select the most voted prediction result.

Growing and voting process of RF:

Bootstrap sample chosen from training set to grow each tree. 2/3 rd for growing each tree. 1/3 rd to calculate OOB error. Random n variables are selected from N samples, n<<N. One variable out of n selected are used at each node to calculate significance of variables in RF classification.

ADVANTAGES OF RANDOM FOREST CLASSIFIER ALGORITHM:

1. The algorithm will avoid overfitting problem.
2. For both classification and regression task the random forest algorithm which is same can be used.
3. Can be used to identify most important features from a training dataset.

They have implemented the model using SUMO traffic simulator. Every second of simulation, vehicle id, speed and location info are taken from sumo. Simulation time is 1300 sec. At each 600th sec accident occurred in defined locations. The accident cases and non-accident cases are entered to ml algorithms as input. And the accuracy of algorithms was compared. 10-fold cross validation method for testing is used. The performance metrics considered are accuracy, sensitivity and specificity.

The results of v2v data confirms that ML algorithms like artificial neural networks, support vector machines, random forests can successfully identify accident using location and speed data. The results of this simulation was used to compare RF algorithm with SVM and ANN. From the results it is found that RF has higher accuracy and sensitivity rate and specificity is comparable to the other two algorithms. It is concluded that RF shows better performance than ANN and SVM at detecting accidents.

ADVANTAGES:

Information about traffic can be gained using vehicles microscopic traffic values. Without V2V equipment deployment, some apps of safety like accident detection can be developed thus reducing the cost of implementation. ML algorithms can be used to analyse vehicle behaviour for detection of unusual behaviors in traffic. These models can be developed without using GPS thus reducing cost.

DISADVANTAGES:

External factors such as pedestrians, traffic signals are not considered. Without using GPS only estimated location is shown.

JOEL DANIEL JOHNSON 19BCT0263

[11] W. J. Chang, L. B. Chen and K. Y. Su, "DeepCrash: A Deep Learning-Based Internet of Vehicles System for Head-On and Single-Vehicle Accident Detection With Emergency Notification," *IEEE*, vol. 7, pp. 148163-148175, 2019.

Deep crash IoV works similar to a dashcam but it provides innovations such as networking connectivity and crash/ accident event recognition functionalities. It provides you with GPS, vehicle speed monitoring, video recording, collision detection, collision event snap, abnormal acceleration detection, connectivity, cloud-based web management platform, web call service, emergency notifications, crash event recognition and more. It can even, to a limited functionality track location in the event of loss of signal of a gps connection.

Deep crash churns through three layers- namely the application layer, network layer and sensing layer.

In the application layer it- Records the information, Employees AI Deep Learning Techniques and manages the platform functionalities.

The networking layer consists of the interfaces, manages communication between the devices and the main server and detect events.
The sensing layer collects the data, obtains in-vehicle information captures crash accident images, while detecting abnormal acceleration and also obtains the location to notify the services in case of emergencies
Algorithm
Step 1: Records and collects data from an unusual event.
Step 2: Sent the event information to the main Deep crash server.
Step 3: Using Deep learning training frameworks, it extracts the features and classify the event
Step 4: If necessary, it then notifies the authorities through web call services.
The proposed IoV system consists of a controller area network bus, Bluetooth, powerline communication networks, V2V networks and V2I networks. These varied networks are interconnected with the utilization of an in-vehicle infotainment platform. Most of these features are already available in the new hybrid vehicles that are available in the market.
It is capable of detecting collisions with an accuracy of 96 percent. When a collision is detected a report is sent to a cloud-based management platform with an average time latency of 7 seconds.

[12] N. Dogru and A. Subasi, "Traffic Accident Detection Using Random Forest Classifier," IEEE, pp. 40-45, 2018.

Traffic accident detection using random forest classifier. In this paper an intelligent accident detection model in which vehicle exchange data and each variable with each other. The data set used is from data collected from VANETs on the basis of speed and coordinates. The model also sends traffic alerts. The model uses machine learning algorithms such as Artificial neural networks, Support vector machine and random forests. The model is implemented using VANET communication devices, Simulation of Urban mobility (SUMO) and machine learning models. Artificial neural networks: ANN is collection of nodes. Transforms weighted sum of i/p to o/p value of 0 or 1. Number of input nodes is equal to number of inputs. And output nodes are dependent on the number of classes. Multi-layer ANN Is called multilayer perceptron. Support vector machine: Recently developed. Accurate and manage large high dimensional datasets. SVMs are used in incident detection, traffic speed and flow predictions, eye movement detection etc. Random Forest: Data mining tool used to solve classification and regression related problems. Random vectors are constructed to grow ensemble of trees and class type. This method improves classification accuracy significantly. Low bias and correlation necessary to achieve higher accuracy

Random Forest basic algorithm

Step1: Selection of random samples from dataset.

Step2: Construct decision tree for each sample. Get prediction result from decision trees.

Step3: Voting is performed for every predicted result.

Step4: Select the most voted prediction result.

Growing and voting process of RF:

Bootstrap sample chosen from training set to grow each tree. 2/3 rd for growing each tree. 1/3 rd to calculate OOB error. Random n variables are selected from N samples, n<<N. One variable out of n selected are used at each node to calculate significance of variables in RF classification.

ADVANTAGES OF RANDOM FOREST CLASSIFIER ALGORITHM:

1. The algorithm will avoid overfitting problem.

2. For both classification and regression task the random forest algorithm which is same can be used.

3. Can be used to identify most important features from a training dataset.

They have implemented the model using SUMO traffic simulator. Every second of simulation, vehicle id, speed and location info were taken from sumo. Simulation time is 1300 sec. At each 600th sec accident occurred in defined locations. The accident cases and non-accident cases are entered to ml algorithms as input. And the accuracy of algorithms was compared. 10-fold cross validation method for testing is used. The performance metrics considered are accuracy, sensitivity and specificity.

The results of v2v data confirms that ML algorithms like artificial neural networks, support vector machines, random forests can successfully identify accident using location and speed data. The results of this simulation were used to compare RF algorithm with SVM and ANN. From the results it is found that RF has higher accuracy and sensitivity rate and specificity is comparable to the other two algorithms. It is concluded that RF shows better performance than ANN and SVM at detecting accidents.

ADVANTAGES:

Information about traffic can be gained using vehicles microscopic traffic values. Without V2V equipment deployment, some apps of safety like accident detection can be developed thus reducing the cost of implementation. ML algorithms can be used to analyses vehicle behavior for detection of unusual behaviors in traffic. These models can be developed without using GPS thus reducing cost.

DISADVANTAGES:

External factors such as pedestrians, traffic signals are not considered. Without using GPS only estimated location is shown.

JOEL DANIEL JOHNSON 19BCT0263

[13] I. A. Zualkeman, F. Aloul, S. A. Qasimi and A. Aishamsi, "DigiMesh-based Social Internet of Vehicles (SIoV) for Driver Safety," *IEEE*, pp. 1-5, 2018.

SIoV enables the driver to be aware of his surroundings and act accordingly. SIoV doesn't need an internet connection to operate. SIoV creates and maintains temporary connections with all its nearby vehicles via a Digimesh wireless ad hoc.
A microcontroller is used in sync with a DigiMesh radio to send and receive bad driving behaviors of their surrounding vehicles.
Dynamic Time Warping is then applied to the data procured by the sensor of each vehicle, creating driver alerts, it can be about speeding, swerving, indications of drowsiness in driving, sudden breaking, etc.
This data can be visible in a LCD with which the driver can interact and be alerted of potential threats. It's been noted that the proposed method can deliver accurate results with accuracy upto 98.6%.
Algorithm

Step 1- includes identifying the borderline behavior which may be- harsh acceleration, harsh braking, turns and/ or swerving
Step 2- this step is repeated every 3 seconds.

- Data collection from OBD II
- Data = low pass filter(data)

Step 3- For each behavior noted in the previous set of behaviors

Step 4-Each s in sample [b] calculate dist(s)= DTW(s, data)
Avg distance(b)= mean of distance(s)

Step 5- Select behavior b’ so that the avg distance (b) = min(avg_dist[b])

Step 6- If b’ is an anomalous behavior, transmit the information to all the vehicles near the area

HARSHA LOKESH 19BCT0240

[14] M. Hassan, S. Mohan, T. Shimizu and H. Lu, "Securing Vehicle-to-Everything (V2X) Communication Platforms," *IEEE,* vol. 5, pp. 693-713, 2020.

Intelligent ground transportation systems have benefited from the development of modern automobile networks. In autos, communication technologies link numerous aspects such as vehicles, people, infrastructures, highways, and cloud computing.

V2X communications makes use of the advantage of the latest networking technology. Vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), vehicle-to-pedestrian (V2P), and vehicle-to-cloud (V2C) connections. The technology of V2X communication is predicted to improve progressively with the changing times as more and more vehicles are connected to the internet.

Even though V2X communication aims to provide a cost efficient and effective transportation infrastructure, V2X technologies which are currently present and also developing ones pose new security challenges. In an example a vehicle where a attacker has penetrated its security can send a false observation about the road like traffic and accident which might or might not have happened and lead other vehicles to believe it is correct, forcing other vehicles to change their behavior by slowing down or rerouting there direction.

IEE802.11p based V2X communication was for interaction with roadside communication and onboard communication (RSUs and OBUs). LTEs allows vehicles to communication with each other by using Uu-based LTE-V2X uplink and downlink channels with base station, and PC5-bases LTE-V2X have sperate sidelink communication.

Security threats to V2X system include Dos attack like jamming, Routing, flooding, sybil, false data injection. Security researches often use attacks that are executed by misbehaving node for security testing. Dos attacks can be performed by using watchdog mechanism i.e., making use of the trust levels of the neighborhood. Packets dropped by some vehicles shows the malicious nature of vehicle and can be blocked by the authorities. Road side units can assist in sybil attack detection using signatures of vehicles in particular time and driving direction, measures according to the previous measured values.

Using real-world field experiments, the simTD project7 examined the role of secure V2X systems to increasing traffic safety and mobility. For the V2X field testing, the team established several concepts, protocols, cryptographic techniques, and privacy-preserving mechanisms. The OVERSEE project suggested a safe and open in-vehicle platform for OEM and non-OEM application execution. RESERVE was one of the most important European projects that tested a variety of V2X security/privacy solutions, with the design and implementation work being submitted to standardization authorities. The goal of the ISE project is to design and build a PKI system that adheres to the ETSI standard. The CAMP (crash avoidance metrics partnership) proposes the detection of misbehavior in the V2X network at local and global network levels.

Integrity can be validated by measuring correctness of event, behavioral/message analysis, location/GPS signal verification, reputation analysis and revocation. System security of vehicular communication needs to be layered to improve resilience for future needs and demands as V2X is the future standard for vehicular safety and standard.

HARSHA LOKESH 19BCT0240

[15] N. Kattukkaran, A. George and M. H. T.P, "Intelligent Accident Detection and Alert System for Emergency Medical Assistance," *IEEE,* pp. 1-6, 2017.

Increase in transportation systems, and vehicular movement has led to increase in number of fatal accidents, mainly due to untimely help from trauma care and emergency ambulance services. Smart alert systems can be put in vehicles for immediate medical aid and help prevent deaths. This system measures acceleration, body sensors for heartbeat to measure victim status to check if medical assistance is crucial.

The system can be implemented by receiving data into mobile application through car and mobile interconnection (Bluetooth/hotspot), which can in succession transmit the critical data with the help of SMS services to the nearest emergency stations for immediate response.

The architecture proposed can say if the vehicle is fallen down i.e., in case of a bike or in an accident where the sudden deceleration is more than normal breaking cases. Accident fall detection using ADXL335 can measure drops in voltage alone three axes', thereby giving accurate detection according to the angle of tilt of vehicles. The heartbeat monitor on wrist MSP430 can measure the sudden changes in heart health, IR LED reflected light can measure change in volume of blood and supply to photodiode which can be used for accurate emergency detection.

Mobile application can compute an accurate response by using an algorithm to detect if vehicle has been in an accident and victim needs immediate attention, according to the data measured such as the angle of tilt, heartbeat.

Alarm/Buzzer and LED light can be used to alert nearby commuters to help or even avoid the victim in the accident, the system sends the accurate data to the emergency service through SMS with the real-time data and monitoring so the victim can be monitored carefully by the time, emergency personals arrive. The implementation of the device can help save lives mainly in rural areas where commuters are less and cannot always be around for help.

HARSHA LOKESH 19BCT0240

[16] F. A. Milaat and H. Liu, "Decentralized Detection of GPS Spoofing in Vehicular Ad Hoc Networks," *IEEE,* vol. 22, pp. 1256-1259, 2018.

The paper proposes a decentralized method for the detection of global-spoof GPS. In this method, vehicles obtain GPS data from their neighboring vehicles by sending them via direct short-range communications (DSRC). The data they collect are then used to generate independent statistical representations of each neighbor. Due to the nature of the civilian GPS signals in the L1, L2 and U2 bands, they are vulnerable to various types of interferences, such as jamming and spoofing. These issues can lead to the exploitation of the GPS signals by unauthorized users.

Statistical models for GPS pseudo-ranges, which are typically tracked by paired vehicles. We find that these pseudo-ranges are affected by non-spoofing events and are not spoofed. We also propose a method that enables vehicle manufacturers to detect spoofed GPS signals. A vehicle utilizes a sum-of-the-parts (CUSUM) procedure to detect anomalous GPS signals. It sends its local detection values to a central cluster head, which uses a modified change-detection procedure to improve its effectiveness. GPS satellites emit carrier signals in the L1 band that contain civilian and military range codes, as well as navigation data such as the satellite's position in orbit, signal transmission time, and satellite clock offset.

The satellite pseudo-ranges are measured by each vehicle and included in the basic safety messages (BSMs) that it transmits to one-hop surrounding vehicles using DSRC. Vehicle's synchronies their pseudo-ranges to the same second of GPS time, we presume. Simulation experiments were used to assess the suggested scheme's performance. Using the pseudo-range error model, 8 vectors of random numbers were produced independently for each vehicle in a cluster for an experiment. Each vector represented a vehicle-estimated pseudo-range to a genuine satellite, with at least four vectors (satellites) detected by two cars in the absence of a spoofing event.

The suggested technique uses the spatial variety and interconnectedness of automobiles to detect faked GPS signals without the need for additional hardware. The experimental results revealed that as the number of cars rose, the average detection delay at the cluster-head dropped, and that the proposed method gradually converged to optimal as the rate of false alarms declined.

HARSHA LOKESH 19BCT0240

[17] B. Paden, M. Cap, S. Z. Yong, D. Yershov and E. Frazzoli, "A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles," *IEEE,* vol. 1, pp. 33-35, 2016.

The Idea of autonomous vehicles with self-driving capability has been around since as early as the 1920s but the basic concept of the idea wasn't put into practice until the 1980s and this was possible due to the pioneering work led by Dr Ernst Dickman's. With the onset of the twenty-first century, this technology has been advancing due to the use of sensing and computing technology and now it is at a point where it can be implemented in real-time. Autonomous vehicles bring in an entirely new dimension to safety, accessibility, efficiency and convenience to automotive transportation and it has started to take its place as an essential requirement for vehicles sold in certain countries such as Europe and the United States of America.

There are many different approaches to autonomous transportation but each approach has to follow a decision-making hierarchy such as route planning, decisions based on the behavior of the vehicle's surroundings, motion planning and vehicle control that make use of low-level and high-level feedback controllers, respectively, that are used in actions such as cruise control and lane-departure systems. Initially, to implement autonomous characteristics to a vehicle, a micro-study approach has to be taken so that the possibilities of errors that occur at the basic level should be sorted out before implementing a macro-study approach and hence the kinematics (Kinematic Single-Track model) of a vehicle in ideal environments should be observed thoroughly and implemented to iron out any ambiguities that may be faced in real-time environments.

Motion planning and trajectory planning layers are required for computing a dynamically feasible trajectory from the vehicle's current configuration to the goal configuration provided by the behavioral layer of the decision-making hierarchy and this is achieved by using machine learning techniques for optimal path-planning that make use of *n-tuple topographs*. These approaches scan the environment and capture a 2-D model of the surroundings of the vehicle and then trains the software to learn and adapt to similar circumstances in the future. While some problem-solving methods can be trivial in real-world scenarios, all the data that is captured is broken down into chunks of raw data and plotted. This plotted data can be used in Probabilistic Road Maps (PRMs) in order to construct roadmaps in high-dimensional configuration spaces because unlike grids, they can naturally run in an anytime fashion and this is used to train a machine to react to normal day-to-day scenarios as well as to dire situations wherein the vehicle occupants may be in danger.

In conclusion, as the autonomy of vehicles improves further, these systems are complex in nature and they need to be broken down into a hierarchy of decision-making problems that are fed as chunks of data into a system. The breakdown into individual decision-making problems has enabled the use of well-developed solution techniques from many areas of research. As we progress to attain higher levels of autonomous driving, we need to find a way to integrate the methods discussed above effectively and efficiently to create robust and dependable machines.

HARSHA LOKESH 19BCT0240

[18] D. Singh and M. Singh, "Internet of Vehicles for Smart and Safe Driving," *IEEE*, pp. 328-329, 2015.

In the paper Smart-Eye solution has been installed into the dashboard of each vehicle. Smart-Eye can store location, driving video, and rear-end collision both before and after the accident. It even stores the vehicle location when there is an accident, which can be a vital piece of information for the concerned authorities. The authorities can get this information from the cloud. There has been significant research going on in this field.

High rise in recorded traffic density, road accidents and crisis faced in regulating effective management of traffic control in urban and rural areas have concerned us to develop an intelligent solution in context to Intelligent Transportation Systems (ITS).

We have achieved a lot of technological advancements in the field of accident management.

However, despite the up-to-date power, control, information, computers and intelligence systems, we face numerous calamities in which thousands of humans die.

ITS is looking to integrate virtual technologies into Vehicles to help prevent the above concerns.

However, this vision is still in its development stage. Currently, Smart-Eye Based solutions play a very crucial role in accident monitoring and prevention.

However, the vision of the intelligent vehicle connectivity stack development on the platform is still.

RAHUL BHASSY 19BCT0099

[19] L.-l. Wang, J.-s. Gui, X.-h. Deng, F. Zeng and Z. f. Kuang, "Routing Algorithm Based on Vehicle Position Analysis for Internet of Vehicles," IEEE, vol. 7, pp. 11701-11712, 2020.

Routing algorithm based on vehicle position analysis for IoV. In this paper vehicle position analysis is used to implement an routing algorithm that is more efficient than 3 other routing algorithms in ratio of packet delivery, delay and routing overhead ratio. The proposed algorithm i.e routing algorithm based on vehicular position (RAVP) is implemented in OMNet++ simulator.

OMNet++ is a simulation environment that is used to compare performance of RAVP with other 3 routing algorithms. The parameter values used in OMNet++ are simulation area, Lane type, Lane interval, Road length, Vehicle number, Vehicle density, Packet size, Data transmission rate, communication radius. The metrics which were considered for performance comparison in OMNeT++ are packet delivery ratio, Average end to end delay and Routing Overhead ratio. After simulation it is noted that the RAVP algorithm outperformed routing algorithms like ICAR, RPUV and R2P.

ADVANTAGES:

After simulation it is noted that the RAVP algorithm outperformed routing algorithms like ICAR, RPUV and R2P in the considered evaluation metrics. RAVP analyses the positional relations of vehicles thus the connectivity of the selected route is better. RAVP algorithm has a lower latency, when the vehicle number is increased, the network connectivity becomes strong. The routing overhead of RAVP is much more stable than other algorithms. It is because the RAVP algorithm gains vehicle movement information by analysis of the position transition probability.

DISADVANTAGES: The RAVP algorithm doesn't consider realistic factors such as pedestrians, traffic and other environmental factors. In this proposed RAVP algorithm network resource utilization is not considered.

RAHUL BHASSY 19BCT0099

[20] R. P. Nayak, S. Sethi and S. K. Bhoi, "PHVA: A Position Based High Speed Vehicle Detection Algorithm for Detecting High Speed Vehicles using Vehicular Cloud," IEEE, pp. 227-232, 2018.

[3] A Position Based High Speed Vehicle Detection Algorithm (PHVA) for Detecting High Speed Vehicles using Vehicular Cloud. In this paper they have implemented PHVA algorithm which detects vehicles moving in high speed in VANET. The adjacent Road side units (RSUs) provide information, based on which the algorithm deals with calculation of speed by the cloud server.

For simulation of PHVA algorithm they have used Vehicles in simulation hybrid framework which uses OMNet++ to set up the network and for road traffic simulation, Simulation of urban mobility is used. The network model mainly contains RSU, roads, traffic, Cloud server and CA. The veins hybrid framework uses IEEE 802.11p and IEEE 1609 for communication. The parameters considered for Simulation Environment are area, no of lanes, types of edges, acceleration, types of vehicles, acceleration, length of vehicles, driver imperfection. The primary parameter to compare PHVA with revise is the detection rate. As the number of vehicles increases the detection rate decreases drastically.

The results indicate that PHVA algorithm has better Accuracy than ReVISE Scheme in the environment.

ADVANTAGES: Accidents can be reduced by detecting high speeds by PHVA algorithm. The PHVA algorithm performs better than ReVISE which is Radio based vehicle detection and speed estimation system. The ReVISE is affected by adjacent physical object as it affects the radio frequency.

DISADVANTAGES: PHVA has good detection till 500 vehicles in the network. But when no of vehicles increases further the detection rate decreases. Vehicle collaboration information is not added in this detection technique. The Efficiency of the detection system can be increased by adding the above-mentioned information.

RAHUL BHASSY 19BCT0099

[21] A. Tigga and A. R. Kumar, "Towards a Vehicle's behavior monitoring and Trust Computation for VANETs," IEEE, pp. 1-6, 2019.

Towards a Vehicle's behavior monitoring and Trust Computation for VANETs. In this paper they have implemented a model for detecting the genuinely of an information sent by other vehicle using a trust and behavior monitoring system using Neuro-fuzzy technique to filter out only legitimate messages.

For simulation of the proposed model, they have used SUMO which runs parallel with OMNet++ using veins framework. Then the neuro fuzzy model is written in c++. The OMNet+= Qtencv GUI makes the network layer transmissions visible. During simulation, veins framework is started then OMNet++ is connected to SUMOs traffic scene using python code via a tcp connection. Each car consists of network interface card, app, mobility rate based TraCL.

The simulation parameters considered are number of nodes, speed, mobility generation, MAC layer, physical layer, propagation model, antenna, communication range of cars, transmission power. In their proposed system Behavior value is also considered which takes into account the external factors like high speed and co2 emission. The evaluation is done on the 30 percent malicious nodes i.e. nodes with high speed and high co2 emission. They have implemented a neuro fuzzy model which is accurate and produce a correct outcome and fast computations. The model clearly differentiates between misbehaving and obedient vehicles. And the messages broadcasted from malicious nodes were discarded.

ADVANTAGES: Clear differentiation between misbehaving vehicles. High detection accuracy and low computational accuracy. Takes factors like co2 emissions into account.

DISADVANTAGES: Lack of external attributes for a more realistic system. Single hop data transmission is used, when multihop transmission across network can also be used.

RAHUL BHASSY 19BCT0099

[22] D. N. Vadhwani and D. Buch, "A Novel Approach for the ITS Application to Prevent Accidents using Wireless Sensor Network, IoT and VANET," IEEE, pp. 1-7, 2019.

A Novel Approach for the Intelligent transportation system Application to Prevent Accidents using Wireless Sensor Network, IoT, VANET in construction area. In this paper they have implemented an accident alert system an application of intelligent transportation system using WSN, IOT and

VANET. They have divided the objectives into accident alert application, Mobile application vehicle infrastructure modification. Two algorithms are implemented for the following objectives.

Assumptions for algorithm: Traffic zone where accidents occur frequently. Place the proposed scenario. Mobile app is installed in the user and vehicle driver mobile. Internet or Wifi is available.

Accident alert and mobile application Algorithm: 1. Monitor vehicles crossing near construction area. If vehicle crosses safe zone to prone area alert or buzzer will be active. Buzzer will be activated when there is chance of accident so vehicle and worker are warned. Distance value from the sensor is sent to microcontroller for alarm activation. Microcontroller post the value to cloud service with the help of wifi. Data received by cloud service will be sent to user mobile app and to the controller for alarm activation. The alert message can be sent to worker or driver via mobile app to warn them in the event of an accident.

They have implemented the proposed system using android app, cloud service, Node MCU-Arduino UNO, Buzzer or alarm, sensors and vehicle or driver. The wireless sensor nodes are used to measure x-acceleration, y-acceleration and z acceleration. The performance metrics was considered on the basis of accuracy in traffic information, time synchronization, synchronization of time in packet detection. The study of different existing models helped to develop a new system model for intelligent transportation system i.e., accident alert system in construction area.

ADVANTAGES:

Volume count measure. Can determine speed of a vehicle, number of vehicles passed.

Accident alert system using android.

DISADVANTAGES:

Cannot monitor live conditions of workers in the construction area. Accuracy in traffic information can still be improved. Unregistered vehicles not included. Accident alert system only works if vehicle is registered with mobile application.

RAHUL BHASSY 19BCT0099

[23] K. A. Kahliq, A. Qayyum and J. Pannek, "Prototype of automatic accident detection and management in vehicular environment using VANET and IoT," IEEE, pp. 1-7, 2017.

Prototype of automatic accident detection and management in vehicular environment using VANET AND IOT. In this paper a model is designed for an automatic accident detection using VANET and IOT. The app is able to detect severity of the accidents and emergency using mechanical and medical sensors. The alert message is sent to the nearest medical center using vanet. The above model is implemented based on a busy city scene When an accident occurs the location and additional details are sent via gps system to nearest hospitals.

The model is implemented using UDOO quad which is used for rapid prototyping, Accelerometer, pulse sensor, muscle sensor, GPS module. To detect Collison the rollover and sudden deceleration of vehicle is detected using accelerometer. After detecting the above-mentioned events the model will check for data collected by biomedical sensors. The system monitors the information if there is a significant difference with normal values, this module generates an alert. Ambulance when arriving also alerts the other vehicles to clear path. Software level implementation is done using c++ socket programming server client model.

The major parts of the system are control room i.e., server-side application, client-side application, vehicle (on board unit) for detecting accidents, Hospital to act upon receiving emergency alert messages. The performance metrics is based on accurate location, fast data transmission, response time, correct information regarding the vitals of accident victims.

Algorithm:

1.Event: Collison/sudden deceleration/roll-over

2. Measure metabolic readings- heartbeat rate/muscle sensor readings from history.

3. if (Yes) significant variance in current metabolic readings and mean metabolic readings

Generate emergency alert

1. (Feedback message) need medical assistance
2. Generate warning alert.

4. if (No) significant variance in current metabolic readings and mean metabolic readings

Discard

They have proposed an application that can auto detect accidents on roads and generate medical service at minimum cost with a centralized system.

ADVANTAGES: Auto detection of accidents on road. Quick alert system. Monitor a person's vital body mechanism as well. Minimum cost for hardware and software. Alert system useful for ambulance to reach destination quickly.

DISADVANTAGES: Testing the accuracy and reliability of communication still required. The IOT system is susceptible to cloud cyber-attacks.

RAHUL BHASSY 19BCT0099

[24] A. Celes and N. E. Elizabeth, "Verification Based Authentication Scheme for Bogus Attacks in VANETs for Secure Communication," IEEE, pp. 0388-0392, 2018.

Verification based authentication for bogus attacks in vehicular ad hoc networks (VANETS) for secure communication. Accurate vehicle to vehicle communication is very important for accident detection. Therefore, security issues and false information's are critical issues that must be dealt with. To prevent these issues cryptography techniques, digital signatures and message verification schemes are used as security measures against VANET attacks. In this paper falsified information's are analyzed primarily with the help of position verification technique. The proposed model is implemented using position-based routing for VANET which includes greedy routing approaches and geographic routing approaches, sensor nodes are used to detect.

Greedy routing the node sends packet to the node which is closest to the final destination. In greedy routing nodes periodically broadcast their positions. In geographic routing every mobile node has built in gps . These features allow locating new and short routes quickly. The sensors used are acceptance range threshold sensor which is an autonomous sensor and is based on max communication range, Proactive exchange of neighbour tables is a cooperative sensor it is used to determine if the received info is true. Reactive position requests sensor is a cooperative sensor which is used to detect nodes which cheats on its position. The proposed system is implemented using NS2 software. The bogus nodes and falsified information is detected by the above mentioned sensors. The performance metrics are based on accurate detection of falsified information and detection of bogus nodes.

In this proposed model false information in the vehicular network is analyzed initially and Sensors detect the nodes that cheated about its position is detected by beacon signals.

ADVANTAGES:

Able to prevent accidents by revealing the nodes that spread false information. Can be implemented in real time scenario

DISADVANTAGES

Use of sensors is limited. Simulation's scenarios need more enhancements. Does not consider external factors like pedestrians, traffic signals etc.

HARSHA LOKESH 19BCT0240

[25] R. Kaur, T. P. Singh and V. Khajuria, "Security Issues in Vehicular Ad-Hoc Network(VANET)," *IEEE*, pp. 884-889, 2018.

Due to the wide range of solutions, it may give, the field of Vehicular Ad-hoc Network (VANET) has attracted a lot of interest and research projects over the years. Information security is regarded as the most important concern in any network system, and this is also true in the VANET. As a result, attackers break the secrecy, privacy, and authenticity features of VANETs wireless conversations between autos, compromising further security. The safety challenges and current threads in the VANET system are presented in this study. This paper focuses on the fundamental security requirements that must be met in order to secure VANETs.

A 3-D Markov technique for providing security in VANETs was presented. In this paper, two types of encryption keys are used: AES-generated keys and Elliptic Curve Cryptography-generated keys. VANETs have been researched for various authentication and validation issues. The author of this research has developed a mechanism to protect user privacy while also addressing the authentication issue.

VANET deployment security needs include, authentication, accessibility, information verification, privacy, event information recording, non-repudiation, tamper evidence hardware, electronic license plates, information correlation, reliability, scalability. Each having its own set of VANET security issues to be fixed.

Attacks on VANETs can be network, application, timing, social, monitoring attack. VANETs play an important role in providing users with safety applications. We have explored several security requirements of users in VANETs in this work. The communication mediums and other elements of VANETs are discussed in this study.

HARSHA LOKESH 19BCT0240

[26] Y. Lin, J. McPhee and N. L. Azad, "Comparison of Deep Reinforcement Learning and Model Predictive Control for Adaptive Cruise Control},," *IEEE*, vol. 6, pp. 221-231, 2021.

In this study it compares Deep Reinforcement Learning (DRL) with Model Predictive Control (MPC) for Adaptive Cruise Control (ACC) design. To approximate the acceleration command dynamics of a vehicle, a first-order system is employed as the Control-Oriented Model (COM). We train a DRL policy using Deep Deterministic Policy Gradient (DDPG) and solve the MPC problem using Interior-Point Optimization based on the control system equations and the multi-objective cost function (IPO).

The findings reveal that, depending on the parameter settings, reinforcement learning solutions can be poorer or better than MPCs in terms of cumulative discounted cost. During testing, the authors also revealed data indicating that reinforcement learning is at least 10 times faster than MPC.

The addition of **ε** is intended to generate a differentiable and smooth cost function, allowing the IPO to converge. At the very least, a pure absolute-value cost function is neither differentiable nor smooth, preventing IPO from converge. The **ε** value is kept low to keep the IPO close to an absolute-value cost while yet allowing it to converge quickly. The popular quadratic cost function for MPC is avoided because it creates considerable steady state errors in DRL solutions. We don't utilize distinct cost functions for DRL and MPC because comparing their episode costs would be pointless.

Case study with initial single condition, control delays with initial different conditions, testing with HFM with constant speed and drive cycles, all give graphs plotted for t[s] and other parameters.

For the comparison, we would include the most recent developments in robust and stochastic MPC, as well as transfer and meta-learning. We might also use environmental noise to train the DRL policy, as it has been found to improve results in tests with stochasticity in the environment.

JOEL DANIEL JOHNSON 19BCT0263

[27] V. Davydov and S. Bezzateev, "Accident Detection in Internet of Vehicles using Blockchain technology," *IEEE,* pp. 766-771, 2020.

This paper proposes two blockchain-based incident discovery models pointing at progressing the ease of law infringement discovery and related measures. They are offline detection blockchain and online detection. Where the offline detection can take place even in case of no internet or communication and will also result in more honestly in road disputes.

There are four rules in particular for the two models.
Rule 1: Each participating vehicle must have an essential block with a few technical data. That square has got to be made by a car merchant quickly after the car was sold, or created by the client himself in case the car is as of now possessed.

Rule 2: Each taking part vehicle must have a vehicle astuteness recognition framework (IRS) and a primitive computing gadget onboard.

Rule 3: There must be as much as conceivable. sync points within the city (stations, street foundation hotspots etc.) for getting data about incidents from vehicle’s possess blockchains.

Rule 4: All recorded incidents must be open-access. In any case, not all data mischances are open and independently settled for each framework.
Online detection models can be formulated on the basis of either a public or private blockchain. Public blockchain- when an accident occurs somebody should verify and update the PoW. The new block is created by the whole network and not just one entity.

Private blockchain- Anybody, who already enlisted within the identifying program, naturally has a get to private blockchain and get keys. When a mishap happens, witnesses around (base stations, cars, people on foot etc.) state the truth and get rewards for confirming. Offline detection model has no connectivity to the roads or to the surrounding units.

In this case the participants must not be greedy and add anything to the blockchain by their own regard. Hence it is necessary to have strong hardware security. Withstanding this, all activities are made by vehicle consequently taking after the calculation, which suggests both members must disable the framework in progress to realize nondisclosure of the mishap.

To conclude, blockchain accident detection trust model which follows proof of work and smart roads facilitating them can detect accidents and make the profiles more trustworthy. The authors also proposed a new offline blockchain model to increase the trust factor provided quality hardware security can be provided.

RAHUL BHASSY 19BCT0099

[28] N. Kumar, D. Acharya, and D. Lohani, "An IoT-Based Vehicle Accident Detection and Classification System Using Sensor Fusion," IEEE Internet Things J., vol. 8, no. 2, pp. 869–880, 2021, doi: 10.1109/JIOT.2020.3008896.

An IoT Based Vehicle Accident Detection and Classification System using Sensor Fusion

In this paper a system which uses the fusion of a mobiles built in and connected sensors to detect and report the accident. The works presents an Iot based automated accident detection and classification system based on the above principal. This method improves the efficiency of search and rescue as it provides the details about type of accident so proper planning, rescue, relief operations according to the situation can be done. In this they have used machine learning models to implement the system , ML model based on Naïve Bayes ,Gaussian mixture model , decision tree techniques .These techniques are then compared to find the best Accident Detection and Classification model .The performance parameters used to test and run the each model related to vehicle motion : speed, absolute linear acceleration, change in altitude ,pitch and roll .These are used to identify what type of accident among collision, rollover, fall-off and no accident etc.

The proposed Iot architecture contains sensor platform (Sensordrone)and a modern smartphone. The sensors present in android smartphone contains sensors such as gps, gyroscope, magnetometer, accelerometer etc. These sensors are used to determine speed, direction, rotation and g-force. Sensordrone send the data to the mobile via Bluetooth. Then the mobile phone processes data which can decrease the internet usage. Then these messages are sent to the IoT server. Then in case of an accident IoT server sends these emergency alert messages with required data to the nearest police station, medical centre, relatives etc.

Variables used speed: A collision or roll over will cause speed to be zero. The functionalities present in the Smartphones GPS are used to determine speed. Absolute linear acceleration: This is used to determine if there is a falloff. Smartphone's accelerometer is used to measure this variable. Change in altitude in one second will determine the falloff. Currently no specialized device is there to detect altitude accurately . Roll and pitch is determined using gyroscope, magnetometer and accelerometer.

Hardware setup used in this paper is Samsung galaxy s8 Android smartphone which has all the above-mentioned sensors. Sensordrone a small sensor hub which is has multiple inbuilt sensors to access environmental variables such as temperature, humidity, light , proximity etc.. It is programmable and have a UART device. In this paper only batrometric altimeter is considered. Software setup used is two android apps called SNUSense, SNUAlertApp. SNUSense to collect and sent users data and SNUAlert app to sent alert message for quick response rescue operation. SNUSense uses the above-mentioned sensors to collect data about the type of accident. SNUAlert app uses google maps with tracker so users can locate the place easily. The smartphone is placed near the centre of gravity of a vehicle to accurately collect data.Rc car is used for experimentation with the above setup.

Using the data received from the above setup they have trained various machine learning models such as Gaussian Mixture Model (GMM) based ADC Model, Decision Tree (DT) based ADC Model. The results indicates that detection system works frequently. and classifies accidents into four classes and reports the incident to defined emergency service, relatives etc. The comparison and results using different models indicate that Naïve Bayes, model is found be highly accurate than other models with 0.95 F1-score.

ADVANTAGES: Uses android mobile as a sensor and for data processing thus minimizing extra costs and internet usage. The proposed system can be fitted to any type of vehicles. Alert system can be predefined to reach relatives, police stations, medical centers.

DISADVANTAGES: The vitals of a car accident victim is not monitored during an accident. The internet connectivity may be disrupted. As the system uses android app to process data and sent alerts the apps can be vulnerable to hackers and other vulnerabilities which may jeopardize the entire system.

HARSHA LOKESH 19BCT0240

[29] T. T. Dandala, V. Krishnamurthy and R. Alwan, "Internet of Vehicles (IoV) for Traffic Mangament," *IEEE*, pp. 1-4, 2017.

In this paper talks about traffic mangament for IoV due to the increase in the number of on road vehicles. Basic idea of traffic monitoring has already been implemented in modern day traffic light systems and management. Pollution checking traffic diviation if there has been an accident or road blockage are some ways that IOT has been used for traffic management.

V2X communication with sensors in the vehicles give onboard data and interconnection between vehicles, helping in security, damage, proximity, tyre pressure alerts, etc. The communication between sensors of vehicles and satelite can also provide emergency services.

Emergency resonse for critical victim in an accident, accident prevention, avoid theft, human detection during night time, traffic control are some of the advantages of using traffic management systems.

PROPOSED SOLUTION

In the paper we provide two model for intelligent accident detection and accident avoidance of Internet of vehicles. For accident avoidance we use a dataset for giving an accurate estimate with Machine learning techniques, to predict and depict accident zones within a few kilometers of range before the accident-prone zone and in turn warn the driver or slow down the car automatically.

The second model we analyze an accident detection dataset for providing accurate estimate of patient condition so that critical help can be provided in time to the accident victim or most critical victim in case of multiple vehicular collision. The data is mainly obtained with the help of various sensors such as biomedical, accelerometer, GPS, etc.

The models provide accurate estimates with lesser fault rate and can help avoid accidents and also provide proper assistance in case of accidents.

IMPLEMENTATION AND RESULT ANALYSIS

The data obtained through the cars Lidar sensors and other IoT sensors can be interpreted and a valid pre-trained model can be used to determine if the vehicle is going to have an accident and the car can be slowed down, or a warning can be given to the driver in order to slow down or stop the car. The model provides accident detection and prevention probabilities and also accident severity for ambulance services and help for respective action to be taken.

VISUALIZATION OF DATA:

POSTED_SPEED_LIMIT	TRAFFIC_CONTROL_DEVICE	WEATHER_CONDITION	LIGHTING_CONDITION	FIRST_CRASH_TYPE	ROADWAY_CONDITION	ROAD_DEFECT	CRASH_TYPE	PRIMARY_CAUSE	MOST_SEVERE_INJURY
30	16	6	1	17	5	5	0	18	1
30	16	1	2	15	0	1	0	22	1
40	4	6	2	2	5	1	0	26	1
30	16	1	3	10	0	1	1	19	1
10	4	1	3	7	0	1	1	26	1

Fig 14. Data set values for model training

The dataset contains 15 parameters for model building, i.e. POSTED_SPEED_LIMIT, TRAFFIC_CONTROL_DEVICE, WEATHER_CONDITION, LIGHTING_CONDITION, FIRST_CRASH_TYPE, ROADWAY_CONDITION, ROAD_DEFECT, CRASH_TYPE, PRIMARY_CAUSE, MOST_SEVERE_INJURY, CRASH_HOUR, CRASH_DAY_OF_WEEK, CRASH_MONTH, LATITUDE, LONGITUDE.

CRASH_TYPE BEING THE TARGET PARAMETER.

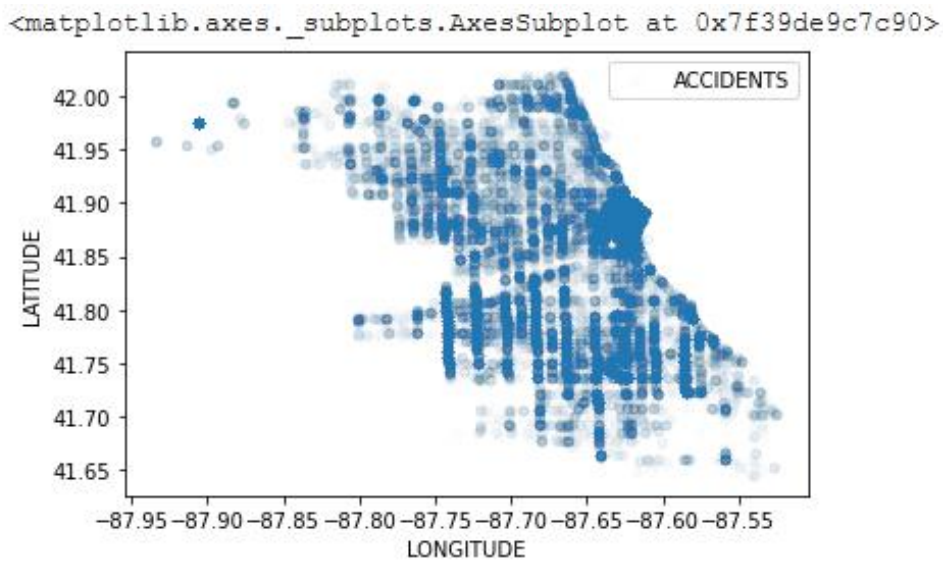


Fig 15. Accident locations, the darker shades tell us a greater number of accidents have taken place at the location

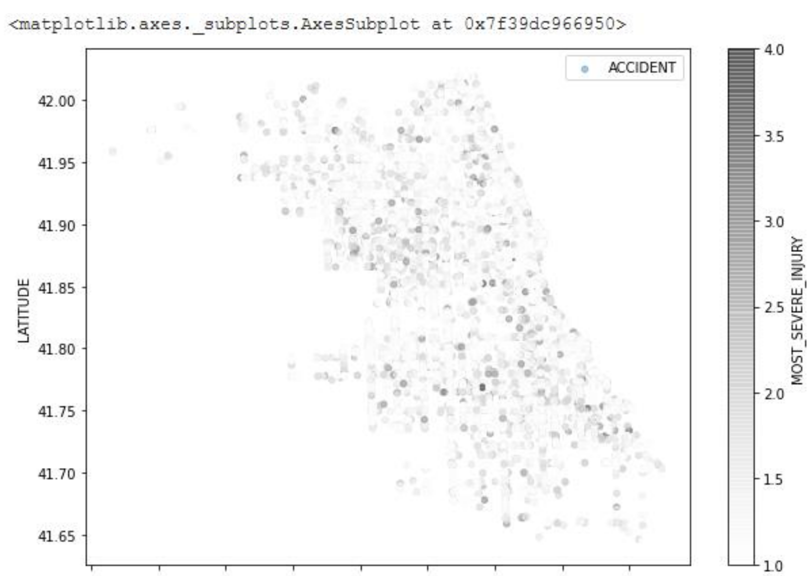


Fig 16. locations with more severe number of accidents

GRADIENT BOOSTING MACHINE

This technique is utilized for regression, classification-based tasks. Boosting comes under the category of ensemble learning technique where it consists of weaker prediction models (which are decision trees) correcting the errors of previous model thus giving more accurate prediction after each model. The process is sequential. This algorithm works well on under sampled data/imbalanced data as every time a weaker model makes an incorrect prediction, the next successive model focuses more that incorrect prediction. But the algorithm needs higher computational demand, making its implementation difficult.

```
from sklearn.ensemble import GradientBoostingClassifier
gbc = GradientBoostingClassifier(loss="deviance", learning_rate=0.1,
      n_estimators=100, subsample=1.0, criterion="friedman_mse",
      min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0,
      max_depth=3, min_impurity_decrease=0.0, init=None,
      random_state=None, max_features=None, verbose=0, max_leaf_nodes=None, warm_start=False)

y_pred = gbc.fit(X_train, y_train.values.ravel()).predict(X_test)
print(format(classification_report(y_test, y_pred)))
print(accuracy_score(y_test, y_pred))
```

```
              precision    recall  f1-score   support

     Serious       0.66      0.61      0.64     10135
      Slight       0.60      0.65      0.62      9078

    accuracy                           0.63     19213
   macro avg       0.63      0.63      0.63     19213
weighted avg       0.63      0.63      0.63     19213

0.6287930047363764
```

Fig 2. Gradient boost classification accuracy for accident severity detection

XGB CLASSIFICATION

It's a version of gradient boosting machines. There are minimal frills in the library since it is intently made for superior computing speed and model execution. It still does, however, provide a lot of sophisticated functions. The gradient boosting decision tree methodology is incorporated in the XGBoost package. Gradient boosting, multiple additive regression trees, stochastic gradient boosting, and gradient boosting machines are all terms used to describe this approach. Gradient boosting is a method in which different models are developed that forecast the covariances or mistakes of previous versions, which are then combined to form the end output. Xgboost is so named since it employs a gradient descending approach to minimize losses while developing newer versions. This method is applicable to both 'regression and classification predictive modeling issues. It is an assembly procedure that requires creating extra models to old models to rectify flaws. Models are created in a logical order until there are no more refinements that can be performed. The two major highlights of XGB classifier is its superior speed and model performance.

```
from xgboost import XGBClassifier
model = XGBClassifier(learning_rate =0.07, n_estimators=300,
                      class_weight="balanced_subsample",
                      max_depth=8, min_child_weight=1,
                      scale_pos_weight=7,
                      seed=27,subsample=0.8,colsample_bytree=0.8)

model.fit(X_train,y_train)

y_pred=model.predict(X_test)

print("Accuracy:",metrics.accuracy_score(y_test, y_pred))
```

```
usr/local/lib/python3.7/dist-packages/sklearn/preprocessing/_label.p
 y = column_or_1d(y, warn=True)
usr/local/lib/python3.7/dist-packages/sklearn/preprocessing/_label.p
 y = column_or_1d(y, warn=True)
ccuracy: 0.48311039400405975
```

```
[ ] print(confusion_matrix(y_test, y_pred))
    print(classification_report(y_test, y_pred))
```

```
[[ 292 9843]
 [  88 8990]]
              precision    recall  f1-score   support

     Serious       0.77      0.03      0.06     10135
      Slight       0.48      0.99      0.64      9078

    accuracy                           0.48     19213
   macro avg       0.62      0.51      0.35     19213
weighted avg       0.63      0.48      0.33     19213
```

Fig 3. XGB classification accuracy for accident severity detection

GUASSIAN NAÏVE BAYES CLASSIFIER

Naïve Bayes algorithms comes within the group of supervised ml algorithms. The algorithm is made using the bayes theorem. The classification technique is simple and provides higher functionality. The algorithm is used to implement many classification problems which are very complex (when dimensionality of input data is very high). Bayes theorem is used in probability for the calculation of conditional probability. The assumption we take while using this technique is that features of the data are strongly independent from each other. Since the features we use in our accident severity dataset are independent, it works well to predict the values for the input set of features. Gaussian naïve bayes comes under naïve bayes techniques. In this model, if the data is continuous, we can assume values tied with each feature/class are distributed according to gaussian (normal) distribution. That is no covariance between dimensions. If the features in the dataset chosen are discrete, we should use multinomial Naïve bayes classification technique for optimal results.

```
] from sklearn.naive_bayes import GaussianNB
nb = GaussianNB()

nb.fit(X_train, y_train)
f1_train = nb.predict(X_train)
f1_test = nb.predict(X_test)

nb_train = metrics.f1_score(y_train, f1_train)
nb_test = metrics.f1_score(y_test, f1_test)
print('Test Accuracy score: ', metrics.accuracy_score(y_test, f1_test))
print('Train F1 score: ', nb_train)
print('Test F1 score: ', nb_test)

Test Accuracy score:  0.8577722946774762
Train F1 score:  0.9092848935820977
Test F1 score:  0.9118514472965593
```

Fig 4. Gaussian naïve bayes classification accuracy for accident prevention

```
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score
gnb = GaussianNB()
y_pred = gnb.fit(X_train, y_train).predict(X_test)
print(accuracy_score(y_test, y_pred))
confusion_matrix(y_test, y_pred)
print(format(classification_report(y_test, y_pred)))

ısr/local/lib/python3.7/dist-packages/sklearn/utils/validat:
 y = column_or_1d(y, warn=True)
.6112007494925311
              precision    recall  f1-score   support

     Serious       0.63      0.63      0.63     10135
      Slight       0.59      0.59      0.59      9078

    accuracy                           0.61     19213
   macro avg       0.61      0.61      0.61     19213
eighted avg       0.61      0.61      0.61     19213
```

Fig 5. Gaussian naïve bayes classification accuracy for accident severity prevention

LOGISTIC REGRESSION

It is one of the fundamental classification techniques. It is the part of linear classifier, and has similarities with polynomial and linear regression. Even though it's a method for binary classification, multiclass problems are also classified with this technique. The probability of categorical dependent variable can be predicted by logistic regression. The assumptions that we take while using logistic regression are categorical dependent variable should be binary. In binary regression desired result is represented as 1 (dependent variable). There should be no covariance/multicollinearity between independent variables. The data set should be quite large. The logistic regression technique measures the dependence of categorical independent variable with other independent variables using a logistic function (sigmoid function) to find their probabilities, which helps in categorizing data into a discrete classis. But we should not use logistic regression technique when no of features is greater than observations as it may lead to overfitting problem. As logistic regression assumes linear dependence between categorical dependent variable and independent variable, it is a major limitation. It should be noted that nonlinear problems cannot be solved with logistic regression.

```
from sklearn.linear_model import LogisticRegression
logisticRegr = LogisticRegression()
logisticRegr.fit(X_train, y_train)
y_pred = logisticRegr.predict(X_test)
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred))
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/utils/validation
  y = column_or_1d(y, warn=True)
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_lo
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver o
    https://scikit-learn.org/stable/modules/linear_model.html#l
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG,
[[6596 3539]
 [3864 5214]]
              precision    recall  f1-score   support

     Serious       0.63      0.65      0.64     10135
      Slight       0.60      0.57      0.58      9078

    accuracy                           0.61     19213
   macro avg       0.61      0.61      0.61     19213
weighted avg       0.61      0.61      0.61     19213
```

Fig 6. Logistic regression classification accuracy for accident severity prevention

K NEAREST NEIGHBOURS

K nearest neighbours basic classification algorithm in machine learning. It is a type of supervised learning techniques. It is used in classification and regression problems as well. A labelled input data is used in supervised learning to train the model and helps in prediction a valid result when unlabeled data is given. This technique does not assume anything about the distribution of data (as some techniques like gaussian Naïve Bayes assume gaussian distribution of the input data). KNN can be used for assigning missing values and resampling input data. K nearest neighbours algorithms learning is based on: Instance based learning, lazy learning and non-parametric. But it is important to keep in mind, there much faster algorithms to produce regression, classification results. KNN does not work with huge dataset, also the used data should not be high dimensional. We should do feature scaling of data before applying K nearest algorithms to get correct predictions.

```
Maximum accuracy:- 0.7859534719774409 at K = 19
```

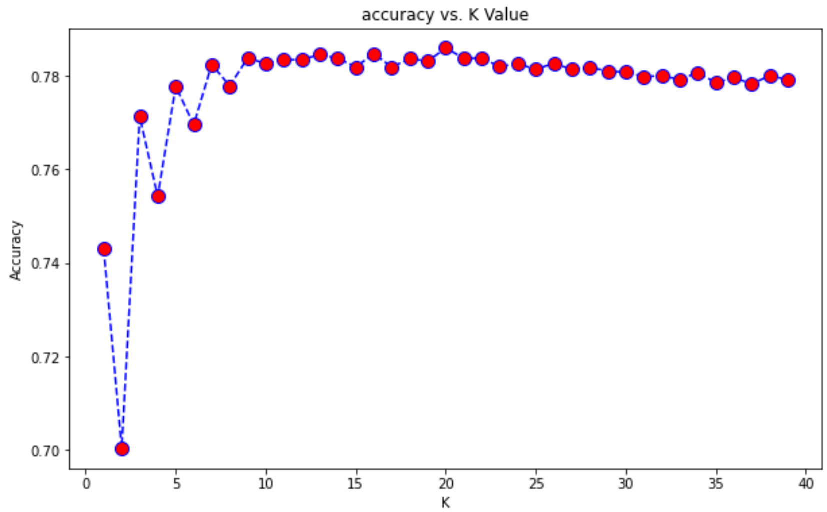


Fig 7. K nearest neighbors classification most efficient K value determination

```python
knn = KNeighborsClassifier(n_neighbors = 19).fit(X_train,y_train)

f1_train = knn.predict(X_train)
f1_test = knn.predict(X_test)

knn_train = metrics.f1_score(y_train, f1_train)
knn_test = metrics.f1_score(y_test, f1_test)
print('Test Accuracy score: ', metrics.accuracy_score(y_test, f1_test))
print('Train F1 score: ', knn_train)
print('Test F1 score: ', knn_test)
```

```
Test Accuracy score:  0.7831335918223475
Train F1 score:  0.8727528399691188
Test F1 score:  0.8653056756608833
```

Fig 8. K nearest neighbours classification accuracy for accident prediction

```python
# import the class
from sklearn.neighbors import KNeighborsClassifier

# instantiate the model (with the default parameters)
knn = KNeighborsClassifier(n_neighbors=3,weights='distance')

# fit the model with data (occurs in-place)
knn.fit(X_train, y_train)

y_pred = knn.predict(X_test)

print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred))
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/neighbors/_classif
  return self._fit(X, y)
[[6306 3829]
 [4427 4651]]
              precision    recall  f1-score   support

     Serious       0.59      0.62      0.60     10135
      Slight       0.55      0.51      0.53      9078

    accuracy                           0.57     19213
   macro avg       0.57      0.57      0.57     19213
weighted avg       0.57      0.57      0.57     19213
```

Fig 9. K nearest neighbours classification accuracy for accident severity detection

RANDOM FOREST CLASSIFICATION

As the name implies, a random forest is made up of a huge number of individual decision trees that work together as an ensemble (Use many learning algorithms to get higher prediction performance than any of the individual learning algorithms could provide). Each tree in the random forest produces a class prediction, and the class with the most votes become the prediction of the model. The key is the low correlation between models. Uncorrelated models can provide ensemble forecasts that are more accurate than any of the individual predictions. The explanation for this amazing effect is that the trees shield each other from their own mistakes. While some trees may be incorrect, many others will be correct, allowing the trees to move in the correct direction as a group. The decision tree is the basic building block of random forest classifiers. A decision tree is a hierarchical structure created from a data set's characteristics. The decision tree is divided into nodes based on a measure connected with a subset of the features. The random forest is made up of a set of decision trees that are linked to a set of bootstrap samples created from the original data set. The entropy of a subset of the characteristics is used to divide the nodes. Using bootstrapping, subsets are formed from the original data set that are the same size as the original data set. The random forest classifier is created with a maximum depth of seven layers and the random state is reset to zero. The random forest is forced to adhere to a simpler model when the depth of the forest is limited. Random forests have the potential to become extremely complex and strong predictive models. It's worth noting that this impurity-based method is prone to noise and can overstate the number of classes in features.

```python
forest_reg = RandomForestClassifier()
forest_reg.fit(X_train, y_train)

f1_train = forest_reg.predict(X_train)
f1_test = forest_reg.predict(X_test)

rfc_train = metrics.f1_score(y_train, f1_train)
rfc_test = metrics.f1_score(y_test, f1_test)
print('Test Accuracy score: ', metrics.accuracy_score(y_test, f1_test))
print('Train F1 score: ', rfc_train)
print('Test F1 score: ', rfc_test)
```

```
Test Accuracy score:  0.8709023616496299
Train F1 score:  0.9999700715290456
Test F1 score:  0.917599415040216
```

Fig 10. Random forest classification accuracy for dataset for accident prediction

```python
from sklearn.ensemble import RandomForestClassifier
#class_weight = dict({2:1, 1:15, 0:50})
rdf = RandomForestClassifier(bootstrap=True,
            class_weight="balanced_subsample",
            criterion='gini',
            max_depth=8, max_features='auto', max_leaf_nodes=Nor
            min_impurity_decrease=0.0,
            min_samples_leaf=4, min_samples_split=10,
            min_weight_fraction_leaf=0.0, n_estimators=300,
            oob_score=True,
            random_state=35,
            verbose=0, warm_start=False)
```

```python
rdf.fit(X_train,y_train)

y_pred=rdf.predict(X_test)
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1:
  """Entry point for launching an IPython kernel.
/usr/local/lib/python3.7/dist-packages/sklearn/base.py:446: User
  "X does not have valid feature names, but"
```

```python
#Import scikit-learn metrics module for accuracy calculation
from sklearn import metrics
# Model Accuracy, how often is the classifier correct?
print("Accuracy:",metrics.accuracy_score(y_test, y_pred))
```

```
Accuracy: 0.6288971009212513
```

Fig 11. Random forest classification accuracy for dataset for accident severity detection

SCALAR VECTOR MACHINE

The "Support Vector Machine" (SVM) is a supervised machine learning technique that can solve classification and regression problems. It is, however, mostly employed to solve categorization difficulties. Each data item is plotted as a point in n-dimensional space, with the value of each feature being the value of a certain coordinate in the SVM algorithm. Then we classify the data by locating the hyper-plane that separates the two groups. Hyperplanes are decision boundaries that aid in data classification. Different classes can be assigned to data points on either side of the hyperplane. The hyperplane's dimension is also determined by the number of features. If there are only two input characteristics, the hyperplane is merely a line. The hyperplane becomes a two-dimensional plane when the number of input features reaches three. When the number of features exceeds three, it becomes difficult to examine. Support vectors are data points that are closer to the hyperplane and have an influence on the hyperplane's position and orientation. We maximize the classifier's margin by using these support vectors. The hyperplane's position will be altered if the support vectors are deleted. The following are some of the benefits of support vector machines: In high-dimensional spaces, it works well. When the number of dimensions exceeds the number of samples, the method is still effective. The goal of SVM is to find a maximum marginal hyperplane (MMH) that splits a dataset into classes as evenly as possible.

```
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.svm import LinearSVC
svm = Pipeline([
("scaler", StandardScaler()),
("linear_svc", LinearSVC(C=1, loss="hinge")),
])
svm.fit(X_train,y_train)
f1_train = svm.predict(X_train)
f1_test = svm.predict(X_test)

svm_train = metrics.f1_score(y_train, f1_train)
svm_test = metrics.f1_score(y_test, f1_test)
print('Test Accuracy score: ', metrics.accuracy_score(y_test, f1_test))
print('Train F1 score: ', svm_train)
print('Test F1 score: ', svm_test)
```

```
Test Accuracy score:  0.8610327811068029
Train F1 score:  0.9111086873534384
Test F1 score:  0.9141768707482993
```

Fig 12. Scalar Vector Machine classification accuracy for accident prevention

Neural Network:

Neural network-based machine learning algorithms don't need to be designed with precise rules that describe what to expect from the input. Instead, the neural net learning algorithm learns by analyzing a large number of labelled instances provided during training and using this answer key to determine which input qualities are required to construct the proper output. After a sufficient number of examples have been processed, the neural network can begin to handle new, unknown inputs and accurately deliver results. Because the computer learns from experience, the more examples and types of inputs it sees, the more accurate the outputs become. Neural networks may be used to solve a wide range of problems and can evaluate a wide range of input types, including photos, videos, files, databases, and more. They also don't necessitate any explicit programming to interpret the contents of those inputs. Because of the generic approach to problem solving that neural network provide, the domains in which this technique can be used are essentially limitless. Image/pattern recognition, self-driving vehicle trajectory prediction, facial identification, data mining, email spam filtering, medical diagnosis, and cancer research are some of the common applications of neural networks nowadays. Today, neural networks are used in a variety of ways, and their popularity is growing rapidly. Each layer of nodes in a deep-learning network trains on a different set of features based on the output of the preceding layer. The following model has been implemented with double layered approach to the neutral network for better accuracy.

```
import tensorflow as tf
from tensorflow import keras
X_train_full, X_test, y_train_full, y_test = train_test_split(X, y)
X_train, X_valid, y_train, y_valid = train_test_split(X_train_full, y_train_full)
print(X_test[:3])
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_valid = scaler.transform(X_valid)
X_test = scaler.transform(X_test)
print(X_test[:3])

input_ = keras.layers.Input(shape=X_train.shape[1:])
hidden1 = keras.layers.Dense(30, activation="relu")(input_)
hidden2 = keras.layers.Dense(30, activation="relu")(hidden1)
concat = keras.layers.Concatenate()([input_, hidden2])
output = keras.layers.Dense(1)(concat)
model = keras.Model(inputs=[input_], outputs=[output])

print(X_train.shape)
print(X_valid.shape)
print(X_test.shape)

model.compile(loss="mean_squared_error", optimizer="sgd",metrics=['accuracy'])
history = model.fit(X_train, y_train, epochs=30,validation_data=(X_valid, y_valid))
mse_test = model.evaluate(X_test, y_test)
```

```
Epoch 28/30
998/998 [==============================] - 5s 5ms/step - loss: 0.1035 - accuracy: 0.8649 - val_loss: 0.1036 - val_accuracy: 0.8647
Epoch 29/30
998/998 [==============================] - 5s 5ms/step - loss: 0.1034 - accuracy: 0.8642 - val_loss: 0.1039 - val_accuracy: 0.8640
Epoch 30/30
998/998 [==============================] - 5s 5ms/step - loss: 0.1033 - accuracy: 0.8649 - val_loss: 0.1032 - val_accuracy: 0.8661
444/444 [==============================] - 1s 3ms/step - loss: 0.1056 - accuracy: 0.8661
```

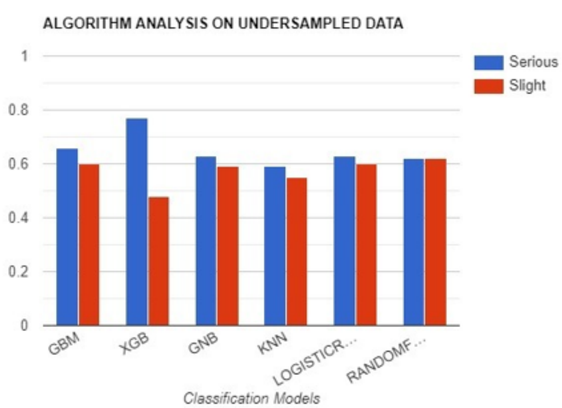
Fig 13. Neutral network method accuracy for accident prevention

Algorithm	Accuracy
GUASSIAN NAÏVE BAYES CLASSIFIER	0.857
K- NEAREST NEIGHBOURS	0.783
SCALAR VECTOR MACHINE	0.861
RANDOM FOREST CLASSIFICATION	0.870
NEUTRAL NETWORK	0.859

Table I: - accident prevention accuracy of classification

Algorithm	Accuracy
GRADIENT BOOSTING MACHINE	0.63
XGB CLASSIFICATION	0.48
LOGISTIC REGRESSION	0.61
K- NEAREST NEIGHBOURS	0.57
GUASSIAN NAÏVE BAYES CLASSIFIER	0.61
RANDOM FOREST CLASSIFICATION	0.62

Table I: - accident severity accuracy of classification



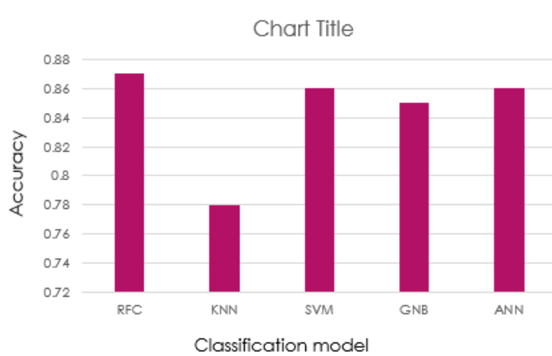


Fig:- Accuracy histograms

CONCLUSION

With this paper we like to comprehensively conclude that the accuracy obtained from various classifications and techniques can be implemented in real life scenarios, the models can be used to train in vehicles in order to prevent accidents and also detect accident severity for emergency services. The prediction or accuracy obtained is compared with other test case dataset in order to determine the most effective data model for the vehicle, the models can be implemented and the model with the least error rate can be used for further betterment and advancement of the technology. Cloud as a service for intelligent vehicle communication can be used for effective APIs for the vehicles in order to warn the drivers in case of accident prevention and send emergency services in case of accident severity detection. The two models are tested for the most accurate result possible for numerical data. The machine learning models are only as effective as the data they are given for implementation, the data for learning can be further modified, cleaning, and better more effective parameters can be given in order for better accuracy result. The paper also gives an analysis of the IoV techniques, such as IoV security measures, VANET communication, vehicle safety, networking, V2X communication, routing. Next, we would like to include various other aspects of Internet of Vehicle transportation and real-life application would be a start.

FUTURE WORKS

Intelligent transportation has a lot of safety, security, privacy, implementation, accuracy concerns still remaining to be addressed, in the future we would like to use more datasets for accurate model implementation and providing the accident detection and accident-avoidance accuracy. The data used can be obtained and trained using real time vehicle sensors rather than using previous datasets with parameters, providing and measuring additional sensors data could help in determining better and faster models. Real world scenarios and live testing of our development would also be one of our many goals. The machine learning models we have implemented can have better build in order to improve the accuracy with the help of better algorithms and classifiers. Improvements on neutral networks can be done in order to improve efficiency. Models can be trained with better GPUs with lesser data lose. The future work we need to address the security concern related to connection vehicles to the Internet as this can have major impact on national security as increase in number of devices connected to the internet can cause increase in number of vulnerability and prone to attacks. The concern about the power requirements and need for clever sources of renewable power in order to run the various sensors for intelligent vehicles also needs to be addressed.

SURVEY TABLE

SERIAL NO	TECHNOLOGY USED	STANDARD/ PROTOCOL/ ALGORITHM	QoS	SIMULATION TOOL/ PLATFORM	YEAR	LIMITATION
[1]	GSM, BLE, RFID, MQ3, Arduino UNO, Raspberry Pi	MQTT, ZIGBEE, NEP, GPS-MNEA	No	none	2020	System is susceptible
[2]	Attacks	Authenticated Key Management, Wizard et al	No	Hardware	2020	Scheme of wazid et al is impractical for deployment in a network
[3]	Microcontroller, Raspberry Pi, RFID, GSM, CPS, CBI, Router	SAD-CS	yes	None	2018	Blockage of signal transmission by mountains, tunnels
[4]	Virtual test environment		No	Microsoft airsim	2019	Condition of malfunctioning
[5]	VANET, IoV, routing	SDN Network architecture, vehicular routing protocol	No	Hardware	2018	Generating high latency due to the calculation on road demand
[6]	IoV, VANET, NS-2	Synthetic VANET mode	No	Omnet++, sumo	2018	Different simulation models for the same model gives different result
[7]	VANET Accelerometer sensors, Gps , gsm AI/ML Accident detection	Diverse	No	Different scenarios. Multiple simulations	2020	VANET's have security and privacy issues.
[8]	SLAM	Single Vehicle SLAM or Multivehicle SLAM Image-to-image, Map-to-map and Image-to-map methods	No	Fast SLAM	2017	No accurate localization algorithm has been formulated. Cost of sensors. Safety of localization algorithm.
[9]	Inertial Measurement Unit	Dead Reckoning	No	KITTI odometry dataset	2019	Accuracy though comparable with visual sensors, is still considerably low and cannot be incorporated into any practical model raw.
[10]	VANET	Destination-Sequenced Distance-Vector (DSDV), Dynamic Source Routing (DSR), and Ad-hoc On Demand Distance Vector (AODV)	No	SWANS, OMNET++	2015	Research should focus on more large scale heterogenous networks for better advancements
[11]	AIoT ADAS Controller Access network Bluetooth	Multisensory decision fusion algorithm	yes	SUMO	2019	Applicable only to the later hybrid vehicles with advanced IoV functionalities.
[12]	Machine Learning	Artificial Neural networks, Support vector machine, Random forest classifier	No	SUMO traffic simulator, Machine learning models.	2018	External factors such as pedestrians, traffic signals are not considered. Without using GPS only estimated location is shown.
[13]	SIoV (DigiMesh ad-hoc wireless network) Microcontroller	Dynamic Time Warping	yes	Hybrid simulation	2018	Location of vehicle is completely unknown. An incident mitigative tool. No action plan in case of an actual incident.
[14]	V2X communication, VANET	LTE, wifi, cellular	yes	OBUs, RSUs,	2020	Security issues regarding V2X are vast and layered defence mechanisms is needed for improvement.
[15]	Biomedical Sensors, ADXL335.	Bluetooth, LTE, GPS, cellular	yes	Cloud application	2017	Effective usage, false rates need to be reduced.
[16]	VANET, security penetration testing	GPS, wifi, cellular	No	MATLAB	2018	Scalability challenges and less effective when vehicle density is less.
[17]	Machine learning algorithms, kinematics, sensors	GPS, wifi, cellular	No	MATLAB	2016	Tailoring and integrating all methods into a valid interaction for implementation
[18]	Smart-Eye application with sensor integration.	Wife, GPS, Cellular, LTE	No	Cloud application, warning system.	2015	Accuracy, security
[19]	Vehicle position analysis and routing.	RAVP algorithm, ICAR ,RPUV, R2P algorithms	yes	Omnet++ simulator	2020	The RAVP algorithm doesn't consider realistic factors such as pedestrians, traffic and other environmental factors. In this proposed RAVP algorithm network resource utilisation is not considered.
[20]	RSU, Cloud technology, Speed detection sensors	IEEE 802.11p and IEEE 1609, PHVA algorithm, Revise Algorithms	No	Simulation Hybrid frame work using OMnet++, Cloud server, Hardware	2018	PHVA has good detection till 500 vehicles in the network. But when no of vehicles increases further the detection rate decreases. Vehicle collaboration information is not added in this detection technique.
[21]	network interface card, Android app, Speed detection and co2 detection sensor	MAC protocol, Neuro fuzzy model using c++,	No	SUMO, OMNet++, c++, Qtencv GUI,python ,Hardware	2019	Lack of external attributes for a more realistic system. Single hop data transmission is used ,When multihop transmission across network can also be used
[22]	WSN, Node MCU-arduino UNO, Wifi ,GSM,GPS	Algorithm for accident alert and mobile app	yes	Android app, Ms azure, Hardware	2019	Unregistered vehicles not included. Accident alert system only works if vehicle is registered with mobile application. Wifi or mobile data may not be available.
[23]	GPS, Accelerometer, Pulse, muscle, Biomedical Sensors, On board units, Vanet	Cellular (3G/4G0, Wifi,	yes	UDOO quad, c++ socket program server client model	2017	Testing the accuracy and reliability of communication still required. The IOT system is susceptible to cloud cyber attacks
[24]	GPS, AUTONOMOUS AND COOPERATIVE SENSORS.	. GREEDY ROUTING TECHNIQUES	no	Ns2, SUMO	2018	Use of sensors is limited. Simulation's scenarios need more enhancements. Does not consider external factors like pedestrians, traffic signals etc
[25]	VANET, V2X, ITS, security testing Dos attacks.	RSUs, OBUs intercommunication	No	Hardware, penetration testing tools.	2018	Security measures not provided
[26]	Deep reinforcement learning, Model predictive control, ML models	None	No	DRL training with data collected	2021	Cost and time constraints.
[27]	Blockchain Smart roads with sync points for the blockchain	Proof of Work	No	Hybrid	2020	Offline blockchains are not secure enough and can be tampered with by the owner
[28]	Machine Learning, Android App, Sensors, wifi, smartphone	Bluetooth, Wifi, Naïve Bayes, Gaussian mixture, Decision Tree models	No	Hardware, Android app based	2021	The vitals of a car accident victim are not monitored during an accident. The internet connectivity may be disrupted. As the system uses android app to process data and sent alerts the apps can be vulnerable to hackers and other vulnerabilities
[29]	VANET, V2X communication	Bluetooth, Wifi, cellular	yes	Hardware, Cloud applicaton,	2017	Security and failure of networks

References

[1] S. Bhuvaneswari and D. R. Saranya, "Internet of Vehicle based Accident Detection and Management Techniques by using VANET: An Empirical Study," *IEEE*, pp. 237-244, 2020.

[2] K. Mahmood, S. Kumari and M. A. Saleem, "Comments on “AKM-IoV: Authenticated Key Management Protocol in Fog Computing-Based Internet of Vehicles Deployment," *IEEE*, vol. 7, pp. 4671-4675, 2020.

[3] S. H. Hadi, A. Saha, F. Ahmad, M. S. Hasan and M. H. Milon, "A Smart Accident Detection and Control System in Vehicular Networks," *IEEE*, pp. 1-6, 2018.

[4] Y. Kang, H. Yin and C. Berger, "Test Your Self-Driving Algorithm: An Overview of Publicly Available Driving Datasets and Virtual Testing Environments," *IEEE*, vol. 4, pp. 171-185, 2019.

[5] L. Alouache, N. Nguyen, M. Aliouat and R. Chelouah, "Survey on IoV routing protocols: Security and network architecture," *WILEY*, vol. 32, no. 2, 2018.

[6] N. M. Adelsamee, S. S. Alsaleh and A. Algarni, "On Simulating Internet of Vehicles," *IEEE*, pp. 1-9, 2018.

[7] U. Alvi, M. A. K. Khattak, B. Shabir, A. W. Malik and S. R. Muhammad, "A Comprehensive Study on IoT Based Accident Detection Systems for Smart Vehicles," *IEEE*, vol. 8, pp. 122480-122497, 2020.

[8] G. Bresson, Z. Alsayed, L. Yu and S. Glaser, "Simultaneous Localization and Mapping: A Survey of Current Trends in Autonomous Driving," *IEEE*, vol. 2, pp. 194-220, 2017.

[9] M. Brossard, A. Barrau and S. Bonnabel, "AI-IMU Dead-Reckoning," *arXiv*, 2019.

[10] J. Cheng, J. Cheng, M. Zhou, F. Liu, S. Gao and C. Liu, "Routing in Internet of Vehicles: A Review," *IEEE*, vol. 5, no. 2339-2352, p. 16, 2015.

[11] W. J. Chang, L. B. Chen and K. Y. Su, "DeepCrash: A Deep Learning-Based Internet of Vehicles System for Head-On and Single-Vehicle Accident Detection With Emergency Notification," *IEEE*, vol. 7, pp. 148163-148175, 2019.

[12] N. Dogru and A. Subasi, "Traffic Accident Detection Using Random Forest Classifier," *IEEE*, pp. 40-45, 2018.

[13] I. A. Zualkeman, F. Aloul, S. A. Qasimi and A. Aishamsi, "DigiMesh-based Social Internet of Vehicles (SIoV) for Driver Safety," *IEEE*, pp. 1-5, 2018.

[14] M. Hassan, S. Mohan, T. Shimizu and H. Lu, "Securing Vehicle-to-Everything (V2X) Communication Platforms," *IEEE,* vol. 5, pp. 693-713, 2020.

[15] N. Kattukkaran, A. George and M. H. T.P, "Intelligent Accident Detection and Alert System for Emergency Medical Assistance," *IEEE,* pp. 1-6, 2017.

[16] F. A. Milaat and H. Liu, "Decentralized Detection of GPS Spoofing in Vehicular Ad Hoc Networks," *IEEE,* vol. 22, pp. 1256-1259, 2018.

[17] B. Paden, M. Cap, S. Z. Yong, D. Yershov and E. Frazzoli, "A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles," *IEEE,* vol. 1, pp. 33-35, 2016.

[18] D. Singh and M. Singh, "Internet of Vehicles for Smart and Safe Driving," *IEEE,* pp. 328-329, 2015.

[19] L.-l. Wang, J.-s. Gui, X.-h. Deng, F. Zeng and Z. f. Kuang, "Routing Algorithm Based on Vehicle Position Analysis for Internet of Vehicles," *IEEE,* vol. 7, pp. 11701-11712, 2020.

[20] R. P. Nayak, S. Sethi and S. K. Bhoi, "PHVA: A Position Based High Speed Vehicle Detection Algorithm for Detecting High Speed Vehicles using Vehicular Cloud," *IEEE,* pp. 227-232, 2018.

[21] A. Tigga and A. R. Kumar, "Towards a Vehicle's behavior monitoring and Trust Computation for VANETs," *IEEE,* pp. 1-6, 2019.

[22] D. N. Vadhwani and D. Buch, "A Novel Approach for the ITS Application to Prevent Accidents using Wireless Sensor Network, IoT and VANET," *IEEE,* pp. 1-7, 2019.

[23] K. A. Kahliq, A. Qayyum and J. Pannek, "Prototype of automatic accident detection and management in vehicular environment using VANET and IoT," *IEEE,* pp. 1-7, 2017.

[24] A. Celes and N. E. Elizabeth, "Verification Based Authentication Scheme for Bogus Attacks in VANETs for Secure Communication," *IEEE,* pp. 0388-0392, 2018.

[25] R. Kaur, T. P. Singh and V. Khajuria, "Security Issues in Vehicular Ad-Hoc Network(VANET)," *IEEE,* pp. 884-889, 2018.

[26] Y. Lin, J. McPhee and N. L. Azad, "Comparison of Deep Reinforcement Learning and Model Predictive Control for Adaptive Cruise Control},," *IEEE,* vol. 6, pp. 221-231, 2021.

[27] V. Davydov and S. Bezzateev, "Accident Detection in Internet of Vehicles using Blockchain technology," *IEEE,* pp. 766-771, 2020.

[28] N. Kumar, D. Acharya and D. Lohani, "An IoT Based Vehicle Accident Detection and classification system using sensor fusion," *IEEE,* vol. 8, pp. 869-880, 2021.

[29] T. T. Dandala, V. Krishnamurthy and R. Alwan, "Internet of Vehicles (IoV) for Traffic Mangament," *IEEE*, pp. 1-4, 2017.

Project dataset / video links: -

Dataset for accident-avoidance model: -
https://data.cityofchicago.org/Transportation/Traffic-Crashes-Crashes/85ca-t3if

Dataset for accident severity model: -
https://drive.google.com/file/d/1k4FMWC96bvpcGA6E3SzCws6fEi0wXRb1/view

Video link for project explanations: -
https://drive.google.com/file/d/1UuqhAyybhF5WqeckU5J42Cgc9OUO5JKW/view?usp=sharing