Encryption Techniques for Intelligent Transportation Systems via Deep Learning for IOV in Smart Cities

Deepak Choudhary¹, Roop Pahuja², Dr. B.R Ambekar National Institute of Technology, Jalandhar, India

ABSTRACT

Now-a-days, there is exponential growth in the field of Wireless Sensor Networks. A connected car is an essential element of the Internet of Vehicles (IoV) vision that is a highly attractive application of the Internet of Things (IoT). The underlying technologies include Internet of Everything (IoE), artificial intelligence, machine learning, neural networks, sensor technologies, and cloud/edge computing. The connectivity between vehicles is through inter communication between sensors and smart devices inside the vehicles, as well as smart systems in the environment as part of the Intelligent Transportation Systems (ITS). In WSN’s security is a major concern, since most of communication happen through a wireless media hence probability of attacks increases drastically. Intrusion detection as well as prevention measures should be taken for secure communication, hence observations of intrusion detection and prevention techniques have taken immense precedence in the research field. With the help of intrusion detection and prevention systems, we can categorize the activities of user in two categories namely normal activities and suspicious activities. There is a need to design effective intrusion detection and prevention system by exploring deep learning for wireless sensor networks. This research aims to deal with proposing algorithms and techniques for intrusion prevention system using deep packet inspection based on deep learning. In this, we have proposed a deep learning model using a convolutional Neural Network classifier. The proposed model consists of two stages like intrusion detection and intrusion prevention. The proposed model learn useful feature representations from a large amount of labeled data and then classifies them. In this work, a Convolutional Neural Network is used to prevent intrusion for wireless sensor networks. To evaluate and test the effectiveness of the proposed system, a WSN-DS dataset is used, and experiments are conducted on the dataset. The experimental results show that proposed system achieves 97% accuracy and performs substantially better than the existing system. The proposed work can be used as a future benchmark for deep learning and intrusion prevention research communities in the smart cities now a days.

Keywords: Wireless Sensor Networks, Deep Learning, Intrusion Detection, Intrusion Prevention Smart cities And Smart Technology, Internet of Vehicles (IOV)

1. INTRODUCTION

Now a day, the internet and the networks are playing a major role in business development and at the same time the network attacks are continuously changing their nature. To deals with network attacks an intrusion detection system (IDS) and intrusion prevention system (IPS) is available. With the help
of an intrusion detection system, we can monitor the network for malicious activity. And when any malicious activity is detected an intrusion detection system alerts the system administrator by generating alarm, but an intrusion detection system is not able to take any action against detected malicious activity [1]. Another solution we have is an intrusion prevention system. Many times peoples think that both intrusion detection and intrusion prevention system are same but IPS is one step ahead than IDS. Basically, intrusion prevention begins with intrusion detection [2]. With the help of IPS, we can detect malicious activity and at the same time detected malicious activity is prevented by giving commands to a firewall. Many organizations are using the traditional tools like firewall, antivirus etc. to secure network but unfortunately these tools are not able to detect unknown attacks. So it is very difficult to achieve the security of networks. In such scenarios, intrusion detection and prevention is the basic requirement to secure networks. Many intrusion detection systems are available such as network based and anomaly based to detect known attacks but the accuracy of same system is low to detect unknown attacks [3]. In network based intrusion detection and prevention system, information is collected and analyzed from the packet itself to detect malicious activity in the network traffic [4]. We needed some techniques to achieve the security of network attacks quickly and accurately. There is a need to secure network from Denial of Service attacks, black hole, grey hole and many more and we can achieve this by designing an intrusion detection and prevention system using deep learning techniques. In last few years, peoples are using deep learning widely for the purpose of classification as well as pattern recognition. While using deep learning, deep learning models are generated by applying many data processing layers. In this paper, we are using the concept of deep packet inspection for the purpose of intrusion detection and prevention in WSN. Basically, deep packet inspection is a form of packet filtering in which header as well as data is examined.

For the purpose of intrusion detection and prevention, few researchers used deep learning techniques. There are many more techniques in deep learning, but among these convolutional neural networks is the effective technique which gives significant performance. By using convolutional neural network, we can achieve best accuracy in the field of intrusion detection and prevention for wireless sensor networks. In this paper, we have used convolutional neural network for intrusion detection and prevention to achieve best accuracy.

The IoT paradigm promises the following benefits:

1. Allows connectivity of all kind of things to develop smarter environments
2. Allows automation to make peoples’ lives easier and more comfortable
3. Enables organisations to automate, increase efficiency and reduce costs
4. Enables companies to cut down on waste and improve service delivery
5. Enables companies to integrate business models and enhance productivity.

With the passing of time, new technologies and mechanisms appeared and matured, e.g. wireless and sensor technologies, machine-to-machine communication, artificial intelligence, machine learning, neural networks, cloud storage, and big data analytics. In the last two decades, the convergence of such technologies and related frameworks has resulted in a number of highly attractive applications of IoT, including the following:

- **Smart Homes**: where status of heating, lighting and security, etc. can be remotely monitored, switched on/off and their operations modified
- **Smart Roads**: where sensory systems embedded in roads warn vehicles of road incidents ahead, and provide information regarding traffic situations
• Smart Healthcare: where technology and smart wearable’s are used as more efficient diagnostic tools, for better treatment of patients
• Smart Manufacturing: where computer-integrated manufacturing results in better adoptability, rapid design changes and intelligent automation
• Industrial IoT: which refers to the implementation of IoT technologies and capabilities in the industrial and manufacturing space
• Intelligent Transportation System: where innovative services relating to traffic management enable safe and smarter use of transport networks
• Internet of Vehicles: where the moving network of IoT enabled vehicles share information collected via in-car smart devices to help decision-making.

In general, IoT touches every industry including manufacturing, agriculture, transportation, logistics, education, healthcare, and retail. Nevertheless, connectivity of devices, especially when these are heterogeneous and their number in the network is increasing with time, imposes numerous new challenges. For example:
• as number of devices increase, amount of information flow increases, which in turn has implications with respect to data safety, security, storage, and data analytics, as well as, decision-making in real-time

The rest of the paper is organized as follows: Section II presents literature survey. Section III presents proposed work. Section IV presents results and discussion by using convolutional neural network to detect and prevent intrusion for WSN. And lastly, section V presents conclusion and future scope.

2. LITERATURE SURVEY

Similar work in the field of intrusion detection and prevention are presented in this section. A discussion on the existing literature related to the topic is presented below.
Leila Mohammadpour et al have proposed a deep learning method to implement an effective and flexible NIDS [5]. In this authors have used a convolutional neural network as an advanced deep learning technique. For implementation purpose authors used the NSL KDD dataset and the proposed system can detect network intrusion with detection rate of 99.97%.
Yuchen Liu et al have proposed a Intrusion Detection Algorithm Based on Convolutional Neural Network [6]. For implementation, the authors used the KDD Cup dataset and the dataset consist of 10 test samples. In this authors have selected samples randomly. Basically, the proposed system can detect attacks with detection rate 97.66% and the proposed system is not able to prevent any type of attacks.
SherazNaseer et al have proposed a deep convolutional neural network-based intrusion detection system [7]. Basically, the proposed system is trained and tested by using NSL KDD dataset and the proposed system can detect anomaly-based attacks. The accuracy of the proposed system is 85.22 %. Again the proposed system is not able to prevent attacks.
AshimaChawla et al have proposed host-based intrusion detection system with a combined convolutional neural network and recurrent neural network [8]. For the purpose of implementation, authors have used the ADFA dataset of system call traces. The proposed system can predict attacks.
ZekiErdem et al have proposed a model network intrusion detection using deep learning [9]. For implementation purposes, authors have used the NSL KDD dataset and the proposed system can prevent zero-day attacks. Basically, to prevent zero-day attacks authors have monitored the system continuously.
FarrukhAslam Khan et al have proposed a novel two-stage deep learning (TSDL) model based on a stacked auto-encoder with a soft-max classifier for efficient network intrusion detection [10]. Basically, the proposed system consists of two stages for the purpose of detection and each stage consists of two hidden layers and a soft-max classifier. In this, each hidden layer is separately pre-trained on unlabeled data. The proposed system is trained and tested using the NSL KDD Cup99 dataset and the detection accuracy of the proposed system is 89.13%. Again the proposed system is not able to prevent any type of attack.

Navaporn et al has proposed an intrusion detection by deep learning with TensorFlow [11]. The proposed can classify different types of attacks and the proposed system requires only packet header information and not the payload. In this, authors have compared their results with a snort and the accuracy of the proposed system is 97%. Based on experimentation, it is concluded that deep learning is a good solution to detect all types of attacks.

Chuanlong Yin et al have proposed a deep learning approach for intrusion detection using recurrent neural networks [12]. The proposed can detect attacks with high accuracy in binary and multiclass classification. For implementation, purpose authors have used the NSL KDD dataset. But the proposed system requires more time for training and in the future, the training time can be reduced.

Wen Hui Lin et al have proposed a model for network intrusion detection for cyber threats using a convolutional neural network [13]. The proposed system can classify multiple threats with a higher detection rate of 95%. The proposed system can be used to enhance the precision of network intrusion detection. Again the proposed system is not able to prevent any types of attacks.

Vinaykumar R et al have proposed a deep learning approach for intelligent intrusion detection systems [14]. The proposed model can detect and classify cyber-attacks. They have evaluated various datasets that are generated over the years through static and dynamic approaches. The proposed model is going to learn the abstract and high dimensional feature representation of the IDS data by passing them into multiple hidden layers. For implementation, purpose authors have used the KDD Cup99 dataset. Monika Roopak et al have proposed deep learning models for cybersecurity in IoT networks [15]. The proposed model doesn't require feature selection to be done before the classification learning and testing but with a large number of attributes in the datasets, the training time could be reduced by applying feature selection before training the model. For implementation, authors have used CICIDS2017 datasets and the proposed system can detect denial of service attack with accuracy of 97.16%.

Amongst the many reasons that exist for the need of intrusion prevention systems, protection from denial of service attacks and protection from many critical exposures are the most important ones. More discussions have been made on intrusion detection system; however it is not effective enough as it deals with intrusions that have already taken effect on the system. When we want more security, then definitely we need strong system and we can achieve this by using a deep learning classifier. In our work, we are using convolutional neural network as a deep learning classifier. Our proposed system will be able to detect as well as prevent intrusion like Denial of Service, Black hole, Gray hole.

**Mobile Ad Hoc Network (MANET)** [2],[42][56][28]

MANET is used for communication between vehicles and roadside smart devices. In this network, also known as Wireless Ad Hoc Network (WANET) or Ad hoc Wireless Network, nodes (i.e. vehicles) are autonomous and mobile in nature and free to move independently in any direction.
Nodes are, therefore, interfaced dynamically in an arbitrary fashion. Hence, the network uses ad hoc nature of routing protocols that are either table-driven or of on-demand variety. MANETs have a self-configuring infrastructure-less network. These have the following features: dynamic topology; variable capacity links; energy constrained operation; and limited physical security. Some MANET are generally restricted to a local area of wireless devices, while others may be connected to the Internet.

In MANET, every node is a potential router. Here, the routing protocols that are often used include: ADOV (Ad Hoc On-demand Distance Vector), DSR (Dynamic Source Routing), OLSR (Optimised Link State Routing) and TORA (Temporally Ordered Routing Algorithm). MANETs are decentralised and, therefore, typically more robust than centralised networks due to the multi-hop nature of data transmission. Since the MANET architecture evolves with time, it has the potential to resolve issues such as isolation or disconnection of nodes or the slowness of Internet data flows. Also, due to the fact that MANET topology changes, these networks are more flexible and scalable, with lower administration costs. However, with the time-evolving nature of the network and changing mobility patterns of vehicles and devices, there can be variations in the network performance and quality of service. There are also concerns relating to signal protection, reliability of nodes, limited processing power, and even the adequacy of power supply. Still, as said before, flexibility of MANET is a huge positive attribute.

**Vehicular Ad Hoc Network (VANET)** [36],[31],[45],[54],[60]

VANET is a subclass of MANET type of connectivity, applicable specifically to vehicles. It consists of groups of moving (and stationary) vehicles, connected via a wireless network. Until recently, VANETs were used mainly to provide safety and comfort to drivers in vehicular environments. Now, however, these networks are seen as infrastructure for an intelligent transportation system (ITS) for an increasing number of autonomous vehicles on the roads. One of the distinguishing characteristics of VANET is the content-centric distribution—the content being important, rather than the source. This is in marked contrast to the Internet where an agent demands information from a specific source. VANET communication protocols are similar to those used in wired networks, where each host has an IP address. However, here, assigning such addresses to moving vehicles is far from trivial and often requires a Dynamic Host Configuration Protocol (DHCP) server, a heresy for Ad Hoc networks that operate without any infrastructure, using self-organisation protocols. One difference between VANET and MANET is that the routing protocols of MANET are not feasible to be used in VANET architecture.

The VANET architecture generally consists of three types of categories viz:

- cellular and WLAN network where fixed gateways and WiMaX/WiFi Aps are used for connection to the Internet for routing and getting traffic information;
- Pure ad hoc i.e. between vehicles and fixed gateways; and
- hybrid i.e. a combination of both infrastructure and ad hoc networks. Various popular architectures of VANET include: WAVE (Wireless Access in Vehicular Environment) by IEEE, CALIM (Continuous Air Interface for Long to Medium range) by ISO, and C2CNet (Car-to-Car Network) by C2C Consortium.
VANETs can be seen as systems consisting of entities that can be divided into three domains as mentioned below: Z. Mahmood

- Mobile domain: that consists of two sub-parts: the vehicle domain (i.e. cars, buses, etc.) and mobile device domain (i.e. handheld devices, smart phones, navigation system, etc.)
- Infrastructure domain: that also comprises two sub-parts: the roadside infrastructure domain (i.e. roadside units and traffic lights) and the central infrastructure domain (i.e. traffic and vehicle management centres)
- Generic domain: that, in turn, consists of Internet infrastructure domain and the private infrastructure domain.

Internet of Vehicles (IoV) [30],[33],[40],[50],[37]

The new era of the IoT is driving the evolution of conventional VANET into the new IoV vision. As such, IoV integrates VANET, IoT and the mobile Cloud Computing. IoT and VANET have been discussed above; the Cloud paradigm provides data storage and data analytics facilities, and provision of all kind of services, whether software related or virtualised hardware. In Vehicle-to-Cloud (V2C) communication, any vehicle in VANET can directly communicate with the cloud environment, and provision cloud-based services. IoV refers to the data interaction, in real time, between vehicles and roadside units using mobile-communication technology, vehicle navigation systems, smart-terminal devices, and information platforms. The objective is to enable information exchange and interaction and a driving–instruction–controlling network system.

IoV can be regarded as an extension of Vehicle-to-Vehicle (V2V) connectivity that enhances driving aids for full autonomous driving by furthering the vehicles’ artificial intelligence awareness of the surrounding environment (including other vehicles and driving-related smart devices). The components of IoV include the following [10]: Vehicles (as nodes of the ad hoc network), Roadside Units (RSU) and infrastructure (e.g. road and traffic related sensors, signals and smart objects), personal devices (e.g. smart phones and PDAs), and humans (e.g. drivers).

Therefore, there are interactions such as the following:

- Vehicle-to-Vehicle interaction
- Vehicle and Infrastructure interaction
- Vehicle and sensors interaction
- Vehicle and RSU interaction
- Vehicle and human interaction
- Personal devices and vehicle interaction
- Cloud and vehicle interaction
- Device-to-device (D2D) interaction.

3. PROPOSED WORK

In this section, the proposed algorithm and the basic concepts of the proposed system are presented. As it is presented in Figure 3.1, the Convolution layer, Max Pooling layer, Dense layer and Flatten layer is used. Figure 3.1 shows the intrusion prevention system architecture using CNN. The stages of the proposed architecture are as follows:
- **Raw Data**
  In this step, we have provided dataset as an input for the purpose of preprocessing.

- **Preprocessing**
  In this step, we have preprocessed input data.

- **Test Data**
  In this step, we can apply test data as an input to the model. Again test data is nothing but the preprocessed data.

- **Convolution Layer**
  Convolution layer is used to extract the features of the input data. In this work Conv1D is used, basically, it is the type of convolution and we can use it for the dataset with sequences.

- **Max Pooling Layer**
  Max pooling layer is used to reduce the size of the representation to reduce the number of parameters as well as computation.

- **Dense Layer**
  A dense layer is used to connect every neuron in one layer to every neuron in another layer. The dense layer uses softmax as an activation function for the classification of generated features of the input.

- **Flatten Layer**
  Flatten layer is used, because we cannot provide rectangular or cubic shapes as a direct input to the system. Again it is in 1-dimensional form.

- **Filtered Data**
  Lastly, we will get filtered data as an output.

![Figure 3.1: Intrusion Prevention System Architecture using CNN](image-url)
Algorithm
The algorithm for the proposed system is presented and discussed below. Initially, we have provided a dataset as an input to a system. Then we have performed preprocessing on accepted input data to achieve the better results from the model. Next, we have trained the model by providing processed data as an input [16][18],[41],[48],[51],[6]. In this, we have considered a Convolutional Neural Network model. Firstly, we have considered the convolution layer of the model. Here we have used Conv1D and basically, it works for sequential datasets [17][19],[31],[45],[54],[60]. For the convolution layer, the size of the kernel is [4 * 4]. Secondly, we have applied the Max Pooling layer and for max-pooling, the size is [2 * 2] and basically, it is used to reduce the size of the representation and to reduce the computation in the network [18]. Thirdly, we have applied the Dense layer and basically, with the help of the Dense layer every neuron in one layer to every neuron in another layer is connected [19][21],[26],[43],[59],[27]. The size of the dense layer is 50 and to prevent the overflow the value for dropout 0.2 is considered. For the purpose of optimization, an Adaptive Moment Estimation (adam) is used and the number of an epoch is set to 10. Again in this, the Rectified Linear Unit (ReLU) is used as an activation function [20][22],[33],[47],[55],[61] and this ReLU uses the Softmax activation function. Fourthly, we have applied the Flatten layer. Basically, we cannot provide input to the system in rectangular or cubic shapes, so we have used the Flatten layer and it is in 1-dimensional form. Then, we have tested the proposed model and for testing, we have provided Test Data as input [23]. Initially, we have considered the value of Attack Type as Null and at the same time, we have initialized the value of Status equal to zero means they all attack type are considered as normal data [24]. Then, we have used to detect function to check the status. If the value of status is equal to zero then the data is considered as normal data and the message “Normal” is printed. If the value of status is not equal to zero then the data is considered abnormal and then we have checked the attack type of that data and the attack type of that data is printed. Lastly, we have evaluated the accuracy and the confusion matrix is printed.

Input: Dataset

Output: Normal, Gray hole, Blackhole, Flooding, TDMA.

Steps:

1. Begin
2. Input: Labeled Training Data as $X = \{X^{(1)}, X^{(2)}, X^{(3)}, \ldots, X^{(c)}\}$, c is the total a number of classes.
3. CNN← X, the raw training data are provided as an input to CNN for feature extraction.
4. $F = \{F^{(1)}, F^{(2)}, F^{(3)}, \ldots, F^{(c)}\}$, the extracted feature vector.
5. Apply Conv1D ( 4, 3, input shape=(3,3), activation=’relu’,padding=’same’)
   Apply Max Pooling (pool size=2, stride=2)
   Apply Dense (50, input_dim= num_features, activation=’relu’)
   Apply Flatten
6. Test the Model, Input= y and display respective attack.
7. Evaluate Accuracy Matrix (y_test, y_pred)
8. Print Confusion Matrix (y_test, y_pred)
9. End
3.1 COMPARASION WITH PREVIOUS WORKS

As the volume of the collected data increases, Deep Learning and Machine Learning (ML) techniques are applied to further enhance the intelligence and the capabilities of an application. The field of smart transportation has attracted many researchers and it has been approached with both ML and IoT techniques. In this, smart transportation covers route optimization, parking, street lights, accident prevention/detection, road anomalies, and infrastructure applications. The purpose of this paper is to make a self-contained review of ML techniques and IoT applications in Intelligent Transportation Systems (ITS) in Smart cities and obtain a clear view of the trends in the aforementioned fields and spot possible coverage needs that there is a great potential in Deep Learning coverage for the Smart Lighting Systems and Smart Parking applications so this proposed techniques is very useful in this applications. Wireless Sensor Networks (WSN), and Information and Communication Technology (ICT) have the potential to address some of the environmental, economic, and technical challenges as well as opportunities in this sector. As the number of interconnected devices continues to grow, this generates more big data with multiple modalities and spatial and temporal variations. Intelligent processing and analysis of this big data are necessary to developing a higher level of knowledge base and insights those results in better decision making, forecasting, and reliable management of sensor.

4. RESULTS AND DISCUSSION

This section gives an idea about the experimentation. In this section, the Convolutional Neural Network classifier is implemented. For implementation, python is used as a scripting language and WSN-DS is used as a dataset. In this experiment, the Convolutional Neural Network is used for classification. The model has been implemented using python and Tensor-Flow on a system equipped with 16GB RAM and CPU Intel Core i7 and Windows 10 system. A WSN-DS dataset consists of 374000 entries and the total entries are divided into different labels as normal, black hole, gray hole, flooding, and TDMA. The proposed system can prevent attacks like a black hole, gray hole, flooding, and TDMA [23]. To check whether data is normal or attack the fields considered are as Is_CH, ADV_S, ADV_R, JOIN_S, JOIN_R, SCH_S, SCH_R, DATA_S, DATA_R, send code [16]. A WSN-DS dataset is shown in Table 4.1.

| id  | Is | wh | A | AD | JO | JO | SC | SC | DA | DA | se | Attac |
|-----|----|----|---|----|----|----|----|----|----|----|----|--------|
| 101 | 1  | 101| 1 | 0  | 0  | 25 | 1  | 0  | 1  | 0  | 120| 0 Norm |
| 101 | 0  | 101| 0 | 4  | 1  | 0  | 0  | 1  | 38 | 0  | 4  | Norm   |
| 206 | 1  | 206| 1 | 2  | 0  | 5  | 1  | 0  | 0  | 676| 0  | Black  |
| 602 | 1  | 602| 1 | 26 | 0  | 0  | 0  | 0  | 0  | 0  | 0  | Black  |
| 211 | 1  | 211| 1 | 11 | 0  | 0  | 0  | 0  | 0  | 0  | 0  | Gray   |
| 602 | 1  | 602| 1 | 26 | 0  | 0  | 0  | 0  | 0  | 0  | 0  | Black  |
| 101 | 1  | 101| 7 | 0  | 0  | 90 | 1  | 0  | 135| 15 | 0  | Flood  |
| 201 | 1  | 201| 8 | 19 | 0  | 1  | 1  | 0  | 238| 238| 0  | Flood  |
| 405 | 1  | 405| 1 | 10 | 0  | 4  | 4  | 0  | 0  | 0  | 0  | TDM    |
| 213 | 1  | 213| 1 | 7  | 0  | 15 | 15 | 0  | 0  | 0  | 0  | TDM    |

The fields of a WSN-DS dataset are as follows-
Id- It is the node id that is used to distinguish the node from another node. Is_CH- It is the flag that is used to distinguish the node as a cluster head or normal node. Who CH- It is considered as a id of the current node. ADV_S- It is considered as a number of messages broadcast by cluster head to the normal node. ADV_R- It is considered as a number of messages received by cluster head from normal node.
JOIN_S- It is considered as a number of join request messages sent from the normal node to the cluster node.

JOIN_R- It is considered as a number of join request messages received by cluster head from a normal node.

SCH_S- It is considered as a number advertise schedule messages sent to the nodes.

SCH_R- It is considered as a number advertise schedule messages received from the cluster head.

DATA_S- It is considered as a number of data packets sent from a normal node to its cluster head.

DATA_R- It is considered as a number of data packets received from the cluster head.

Send Code- It is considered as a cluster sending code.

Attack Type- It is considered as a type of attack or simply labels.

Normal:
To check whether the data transmitted is normal following two cases are considered.
Case I:
If the node is a cluster head then we have assigned value $\text{Is}_\text{CH}=1$. Then this node is going to broadcast an advertising message with value $\text{ADV}_\text{S}=1$ and at the same time, the value of $\text{ADV}_\text{R}$ is null. After sending an advertising message this node receives join requests from other nodes like $\text{JOIN}_\text{R}$ and then sends scheduling message $\text{SCH}_\text{S}$. Lastly, we have considered some value for $\text{DATA}_\text{R}$ and $\text{DATA}_\text{S}$ and send code is considered with the value null.

Case II:
If the node is not a cluster head then we have assigned value $\text{Is}_\text{CH}=0$. Then this node is going to receive an advertising message with value $\text{ADV}_\text{R}=4$ and at the same time, the value of $\text{ADV}_\text{S}$ is null. After receiving an advertising message this node sends join requests to other nodes like $\text{JOIN}_\text{S}$ and then sends scheduling message $\text{SCH}_\text{R}$. Lastly, we have considered some value for $\text{DATA}_\text{R}$ and $\text{DATA}_\text{S}$ and send code is considered with the value null.

If the system can satisfy the above two cases then we can say that the transmitted data belongs to the category of the normal type. Both the cases for the normal category are shown in Table 4.2.

Table 4.2: Cases for Normal category

| Id  | Is C | who | ADV_S | ADV_R | JOIN_S | JOIN_R | SCH_S | SCH_R | DATA_S | DATA_R | send_co | Attack  |
|-----|------|-----|-------|-------|--------|--------|-------|-------|--------|--------|---------|---------|
| Cas 10100 1 10100 1 0 0 25 1 0 0 1200 0 0 Norm |
| Cas 10100 0 10104 0 4 1 0 0 1 38 0 4 4 Norm |

Blackhole:
In the blackhole, the attacker node receives data from the source node and the received data is not forwarded to the destination node [24]. Simply attacker node drops all packets.

To check whether the data transmitted is normal or not the following two cases are considered.

Case I:
If the node is cluster head we have assigned value $\text{Is}_\text{CH}=1$, then this node is going to broadcast advertising message to another node as $\text{ADV}_\text{S}=1$ and at the same time is going to receive advertising requests from other nodes as $\text{ADV}_\text{R}=2$. Again in this, we have considered the value of $\text{JOIN}_\text{S}$ is equal to null as $\text{JOIN}_\text{S}=0$ and the value of $\text{JOIN}_\text{R}$ is equal to non-null as $\text{JOIN}_\text{R}=2$. Then this node will send scheduling messages as $\text{SCH}_\text{S}=1$ and receives data message as $\text{DATA}_\text{R}=5$. And lastly, the value of send code $=0$.

Case II:
In this case, actually, the node is not the cluster head but the same node is telling identity as a cluster head. So we have considered the value of this node as $\text{Is}_\text{CH}=1$. Then this node is going to broadcast advertising messages to another node as $\text{ADV}_\text{S}=1$ and at the same time is going to receive advertising requests from other nodes as $\text{ADV}_\text{R}=26$. Again in this, we have considered the value of $\text{JOIN}_\text{S}$, as well as $\text{JOIN}_\text{R}$, which is equal to null as $\text{JOIN}_\text{S}=0$ and $\text{JOIN}_\text{R}=0$ respectively. Again in this the value of $\text{DATA}_\text{R}$ and send code is considered as null as $\text{DATA}_\text{R}=0$ and send code $=0$.

Both the cases for the blackhole category are shown in Table 4.3.
Table 4.3: Cases for Blackhole category

| id  | Is_C | who | ADV | ADV | JOIN | JOIN | SCH | SCH | DATA | DATA | send_co | Attack |
|-----|------|-----|-----|-----|------|------|-----|-----|------|------|--------|--------|
| Cas20602 | 1    | 20602 | 1   | 2   | 0     | 5    | 1   | 0   | 0    | 676   | 0       | Blackho |
| Cas60200 | 1    | 60210 | 1   | 26  | 0     | 0    | 0   | 0   | 0    | 0     | 0       | Blackho |

**Gray hole:**
In the gray hole, the attacker node receives data from the source node and the received data is not forwarded to the destination node [25][26],[27]. Simply attacker node drops some packets.

To check whether the data transmitted is normal or not the following two cases are considered.

**Case I:**
If the node is cluster head we have assigned value Is_CH=1, then this node is going to broadcast advertising message to another node as ADV_S=1 and at the same time is going to receive advertising requests from other nodes as ADV_R=2. Again in this, we have considered the value of JOIN_S is equal to null as JOIN_S=0 and the value of JOIN_R is equal to null as JOIN_R=0. Then all other values are considered as scheduling messages as SCH_S=0 and receives data message as DATA_R=0. And lastly, the value of send code =0.

**Case II:**
In this case, actually, the node is not the cluster head but the same node is telling identity as a cluster head. So we have considered the value of this node as Is_CH=1. Then this node is going to broadcast advertising messages to another node as ADV_S=1 and at the same time is going to receive advertising requests from other nodes as ADV_R=26. Again in this, we have considered the value of JOIN_S, as well as JOIN_R, which is equal to null as JOIN_S=0 and JOIN_R=0 respectively. Again in this the value of DATA_R and send code is considered as null as DATA_R=0 and send code=0.

Both the cases for the grayhole category are shown in Table 4.4.

Table 4.4: Cases for Grayhole category

| id  | Is_h | wh | A | A | JOIN | SC | SC | DA | DA | se | Att |
|-----|------|----|--|---|-----|----|----|----|----|----|-----|
| Cas20602 | 1    | 21 | 1 | 21 | 1   | 11  | 0   | 0   | 0   | 0   | 0   | Gr   |
| Cas60200 | 1    | 60 | 1 | 60 | 1   | 26  | 0   | 0   | 0   | 0   | 0   | Gr   |

**Flooding:**
Basically in flooding attacks, the bogus message is generated to increase the network traffic for consuming server or network resources. To check whether the data transmitted is normal or not the following two cases are considered.

**Case I:**
If the node is a cluster head then we have assigned value Is.CH=1. Then this node is going to broadcast an advertising message with value ADV_S=1 and at the same time, the value of ADV_R is null. After sending an advertising message this node receives join requests from other nodes like JOIN_R =90 and then sends scheduling message SCH_S. Lastly, we have considered some value for DATA_S=1350 and DATA_S= 15 and the value of send code is considered as null.

**Case II:**
If the node is a cluster head then we have assigned value Is.CH=1. Then this node is going to broadcast an advertising message with value ADV_S=7 and at the same time, the value of ADV_S is non-null. After broadcasting advertising message this node receives join requests from other nodes like JOIN_R and then sends scheduling message SCH_S. Lastly, we have considered some value for DATA_S= 238, DATA_S= 238 and the value of send code is considered as null. Both the cases for the flooding category are shown in Table 4.5.
Table 4.5: Cases for Flooding category

| Id | Is | wh | A | A | JO | JO | SC | SC | D | D | se | Att |
|----|----|----|---|---|----|----|----|----|---|---|----|-----|
| Ca | 101 1 | 101 7 | 0 | 0 | 90 | 1 | 0 | 13 | 15 | 0 | Flo |
| Ca | 201 1 | 201 8 | 19 | 0 | 1 | 1 | 0 | 23 | 23 | 0 | Flo |

**TDMA:** Time Division Multiple Access is one type of channel access technique that is used for shared medium networks. With the help of TDMA, many users can share the same frequency channel by dividing the signals into different time slots[25]. Both the cases for the TDMA category are shown in Table 4.6

Table 4.6: Cases for TDMA category

| Id | Is | wh | AD | A | J | J | SC | S | DA | DA | sen | Att |
|----|----|----|----|---|---|---|----|---|----|-----|-----|-----|
| Ca | 40 1 | 40 1 | 10 | 0 | 4 | 4 | 0 | 0 | 0 | 0 | 0 | TD |
| Ca | 21 1 | 21 1 | 7 | 0 | 15 | 15 | 0 | 0 | 0 | 0 | 0 | TD |

The Table 4.7 shows the details of the packet transmitted as a normal by using our proposed algorithm.

Table 4.7: Details of Normal Data

| id | Is | who | A | A | JO | JO | SC | S | DA | D | Dat | se | Att |
|----|----|-----|---|---|----|----|----|---|----|---|-----|----|-----|
| 101 1 | 101 1 | 0 | 0 | 25 | 1 | 0 | 0 | 12 | 48 | 0 | No |
| 104 0 | 104 0 | 1 | 1 | 0 | 0 | 1 | 13 | 0 | 0 | 1 | No |
| 101 0 | 101 0 | 4 | 1 | 0 | 0 | 1 | 181 | 0 | 0 | 3 | No |
| 110 0 | 110 0 | 7 | 1 | 0 | 0 | 1 | 43 | 0 | 0 | 3 | No |
| 270 0 | 270 0 | 0 | 0 | 0 | 0 | 0 | 13 | 11 | 13 | 0 | No |
| 209 0 | 209 0 | 4 | 1 | 0 | 0 | 1 | 72 | 0 | 0 | 4 | No |
| 112 0 | 112 0 | 6 | 1 | 0 | 0 | 1 | 80 | 0 | 0 | 5 | No |
| 304 0 | 304 0 | 7 | 1 | 0 | 0 | 1 | 96 | 0 | 0 | 6 | No |
| 101 0 | 101 0 | 5 | 1 | 0 | 0 | 1 | 51 | 0 | 0 | 1 | No |
| 101 0 | 101 0 | 7 | 1 | 0 | 0 | 1 | 51 | 0 | 0 | 5 | No |

The table 4.8 shows the details of the attacks prevented using our proposed algorithm. By using our proposed algorithm all mentioned types of attacks are prevented and shown in the table 4.8.

Table 4.8: Attacks Prevented using CNN

| id | Is | who | A | A | JO | JO | SC | S | DA | DA | DA | DA | sen | Att |
|----|----|-----|---|---|----|----|----|---|----|-----|-----|-----|-----|-----|
| 211 1 | 2110 1 | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Gra |
| 405 1 | 4050 1 | 10 | 0 | 4 | 4 | 0 | 0 | 0 | 0 | TD |
| 602 1 | 6021 1 | 26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Blac |
| 602 1 | 6021 1 | 26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Gra |
| 604 1 | 6040 1 | 27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Gra |
| 170 1 | 1702 1 | 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Nor |
| 404 1 | 4041 1 | 26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Gra |
| 906 1 | 9060 1 | 23 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Gra |
| 213 1 | 2130 1 | 5 | 0 | 4 | 4 | 0 | 0 | 0 | 0 | TD |
| 206 1 | 2060 1 | 2 | 0 | 5 | 1 | 0 | 0 | 676 | 0 | Blac |
| 103 1 | 1030 1 | 5 | 0 | 32 | 1 | 0 | 0 | 124 | 0 | Blac |
| 213 1 | 2131 1 | 7 | 0 | 15 | 15 | 0 | 0 | 0 | 0 | TD |
The table 4.9 shows the confusion matrix, it gives the value of precision, recall, F1-score, and accuracy. The overall accuracy of our proposed algorithm is 97%.

Table 4.9: Confusion Matrix

|          | Precision | Recall | F1-Score |
|----------|-----------|--------|----------|
| Normal   | 0.99      | 0.99   | 0.99     |
| Blackhole| 0.94      | 0.90   | 0.92     |
| Grayhole | 0.89      | 0.81   | 0.85     |
| Flooding | 0.73      | 0.50   | 0.59     |
| TDMA     | 0.63      | 0.93   | 0.75     |
| Accuracy | 0.97      | 0.97   | 0.97     |

Figure 4.1: Graphical Representation of Attacks Prevented using CNN

The figure 4.1 shows the graphical representation of attacks prevented using CNN. In the figure, precision, recall, and F1-score are given for attacks prevented using CNN. Accuracy for Normal, Blackhole, Grayhole, Flooding and TDMA are 99%, 94%, 89%, 73% and 63% respectively.

Figure 4.2: Attacks prevented using CNN with different data size
The figure 4.2 shows the results of attacks prevented using CNN with different data sizes. The figure shows the results of Precision, Recall and F1-Score for different data sizes. The proposed model is tested on the same dataset but every time size of the dataset considered is different. Based on results obtained, we can say that if we vary the size of the dataset the accuracy remains the same so that the size of data is not going to affect the proposed model.

5. CONCLUSION AND FUTURE SCOPE

From the discussions in the above sections, we understand the need to design a system that can prevent intrusions in a WSN. Due to anything, anytime, anywhere, the type of computing use of WSN has increased significantly. Looking at the various threats and several attacks in a wireless network, and security is the prime concern. Intrusion is one of the critical issues, and intrusion detection has already reached its saturation. Therefore, we need a scalable and attack resistance proactive intrusion prevention system using deep packet inspection in a wireless sensor network. Artificial Intelligence (AI), sometimes called Machine Intelligence, refers to intelligence demonstrated by machines. The term is often used to describe machines (e.g. computers and cars) that mimic cognitive functions that humans associate with learning and problem solving. Machine intelligence is gained by performing statistical operations on data to discover patterns therein, rather than intelligence being explicitly programmed within the systems. AI can be categorized into three different types, as mentioned below:

- Analytical: this has characteristics consistent with cognitive intelligence, including learning based on prior experience
- Human-inspired: this has elements from cognitive and emotional intelligence that contribute to the decision-making processes.
- Humanised: this has characteristics of several different types of competencies (e.g. cognitive, emotional, social, etc.) and has the ability to be self-aware.

Machine Learning (ML), sometimes called Deep Learning, is the application of AI. It enables a system to automatically learn and improve with experience, without human intervention, beyond the initial programming that sets up the learning process. ML methods include: supervised ML algorithms, unsupervised ML algorithms, semi-supervised ML algorithms, and reinforcement ML algorithms. Somewhat-known classes of algorithms include: genetic algorithms, rule-based ML, learning classifier systems, similarity and metric learning, clustering algorithms, Bayesian algorithms, neural networks, decision tree learning, etc. ML is currently the most promising path to strong AI [18] [29],[39],[44],[59],[51]. In the context of connected vehicles, some useful features of AI-based systems include speech recognition, driver monitoring, virtual driver assistance, camera-based vision systems, and radar-based detection (of other vehicle and roadside units) systems.

In this paper, Convolutional Neural Network algorithm is implemented to detect and prevent intrusion for Wireless Sensor Network. For implementation purposes, a WSN-DS dataset is used as an input to our system. At the time of implementation, different size of the dataset is used, and based on that, we can conclude like if we vary the size of a dataset, and accuracy of our algorithm not gets affected. In this paper, the results of implementation using Convolutional Neural Network with different data sizes are presented. So, Convolutional Neural Network algorithms are better to detect and prevent intrusion for WSN. Through the literature survey, we understand that there is a need to develop a scalable and attack resistance system for intrusion prevention using deep packet inspection in a WSN. A system is proposed to detect and prevent intrusion using the Convolutional Neural Network for the WSN and the accuracy of intrusion prevention with our system is 97%.

Hence, the proposed system can be used to detect and prevent attacks in all types of networks where security matters.
REFERENCES

[1] Deepak Choudhary, Roop Pahuja “Prevention of Intrusion attacks via Deep Learning Algorithm in Wireless Sensor Network in Smart Cities” in Innovations in Clean Energy Technologies(ICET-2020), 27th-28th August 2020.

[2] Deepak Choudhary, Roop Pahuja “Performance Optimization by MANET AODVDTNCommunication” in SCOTA 2019 held on dec2019 in NIT patna Spring scopous indexed 2019.

[3] Pankaj R Chandre, Parikshit N Mahalle et al “Deep Learning and Machine Learning Techniques for Intrusion Detection and Prevention in Wireless Sensor Networks: Comparative Study and Performance Analysis” Design Frameworks for Wireless Networks, 95-120, Springer, Singapore, 2020.

[4] Nilanjan Dey, Parikshit N Mahalle et al “Applied Machine Learning for Smart Data Analysis” CRC Press, 2019.

[5] Leila Mohammadpour, Teck Chaw Ling, Chee Sun Liew and Chun Yong Chong “A Convolutional Neural Network for Network Intrusion Detection System”, Proceedings of the APAN – Research Workshop 2018, ISBN 978-4-9905448-8-1.

[6] Yuchen Liu, Shengli Liu & Xing Zhao “Intrusion Detection Algorithm Based on Convolutional Neural Network”, State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou, Henan, China.

[7] Sheraz Naseer1, and Yasir Saleem “Enhanced Network Intrusion Detection using Deep Convolutional Neural Networks” KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS VOL. 12, NO. 10, Oct. 2018.

[8] Ashima Chawla, Brian Lee, Sheila Fallon and Paul Jacob “Host based Intrusion Detection System with Combined CNN/RNN Model” Athlone Institute of Technology, Ireland.

[9] U˘gurÇekmez, Zeki Erdem, Ali Gökhan Yavuz, O¨zgür Koray Sahingo¨z, Ali Buldu “DerinÖ˘grenme ile A˘gAnomaliTespiti Network Anomaly Detection with Deep Learning” 978-1-5386-1501-0/18/$31.00 ©2018 IEEE.

[10] Farrukh Aslam Khan, Senior Member, IEEE, Abdu Gumaei, Abdelouahid Derhab, and Amir Hussain “TSDL: A Two-Stage Deep Learning Model for Efficient Network Intrusion Detection “DOI 10.1109/ACCESS.2019.2899721, IEEE Access.

[11] Navaporn Chockwanich, Vasaka Visoottiviseth “Intrusion Detection by Deep Learning with Tensor Flow” International Conference on Advanced Communications Technology (ICACT), ISBN 979-11-88428-02-1.

[12] Chuanlong Yin, Yuefei Zhu, Jinlong Fei, And Xinzeng He “A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks “ 2169-3536 2017 IEEE. Translations, VOLUME 5, 2017.

[13] Wen-Hui Lin1, Hsiao-Chung Lin, Ping Wang, Bao-Hua Wu, Jeng-Ying Tsai “Using Convolutional Neural Networks to Network Intrusion Detection for Cyber Threats “Proceedings of IEEE Internatona Conference on Applied System Innovation 2018, IEEE ICASI 2018- Meen, Prior & Lam (Eds).

[14] Vijaykumar R et al “Deep Learning Approach for Intelligent Intrusion Detection System”, 10.1109/ACCESS. 2019.2895334, IEEE Access.

[15] Monika Roopak et al “Deep Learning Models for Cyber Security in IoT Networks”, 978-1-7281-0554-3/19/$31.00©2019 IEEE.

[16] Sachin Babar, Parikshit Mahalle et al “Proposed security model and threat taxonomy for the Internet of Things (IoT)”, International Conference on Network Security and Applications, 420-429, Springer, Berlin, Heidelberg, 2010.

[17] Tianyi Liu, Shuang Fang et al “Implementation of Training Convolutional Neural Networks” arXiv:1512.07108v6 [cs.CV] 19 Oct 2017.
[18] Jiuxiang Gu, Zhenhua Wang et al. “Recent Advances in Convolutional Neural Networks” arXiv: 512.07108v6 [cs.CV] 19 Oct 2017.

[19] Gao Huang, Zhuang Liu et al. “Densely Connected Convolutional Networks” arXiv: 1608.06993v5 [cs.CV] 28 Jan 2018.

[20] Vinod Nair, Geoffrey E. Hinton “Rectified Linear Units Improve Restricted Boltzmann Machines” Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel, 2010.

[21] Aakash Atul Alurkar, Sourabh Bharat Ranade et al. “A Comparative Analysis and Discussion of Email Spam Classification Methods Using Machine Learning Techniques” Applied Machine Learning for Smart Data Analysis, 185, CRC Press, 2019.

[22] Parikshit N Mahalle, Bayu Anggorojati et al. “Identity authentication, and capability based access control (iacac) for the internet of things”, Journal of Cyber Security and Mobility, 1, 4, 309-348, 2013.

[23] Iman Almomani, Bassam Al-Kasasbeh, and Mousa Al-Akhras. “WSN-DS: A Dataset for Intrusion Detection Systems in WSNs” Hindawi Publishing Corporation Journal of Sensors Volume 2016, Article ID 4731953.

[24] Umashankar Ghugar, Dr. Jayaram Pradhan. A Study on Black Hole Attack in Wireless Sensor Networks International Journal of Advance Computing Technique and Applications (IJACTA), ISSN : 2321-4546, Vol 5, Issue 1, January- June, 2017.

[25] Arvind Dhaka, Amita Nandal et al. “Gray and Black Hole Attack Identification using Control Packets in MANETS” Eleventh International Multi-Conference on Information Processing-2015 (IMCIP-2015), Procedia Computer Science 54 (2015) 83–91.

[26] Nipa Patan, Rajan Patel. “A Mechanism for Prevention of Flooding based DDoS Attack” International Journal of Computational Intelligence Research ISSN 0973-1873 Volume 13, Number 1 (2017), pp.101-111© Research India Publications http://www.ripublication.com.

[27] Michael Manzo, Tanya Roosta et al. “Time Synchronization Attacks in Sensor Networks” SASN’05, November 7, 2005, Alexandria, Virginia, USA. Copyright 2005 ACM 1-59593-227-5/05/0011.

[28] Technavio (2019) Top 10 connected car companies leading the global connected car market in 2019. https://blog.technavio.com/blog/top-10-connected-car-companies-leading-global-connected-

[29] Business Insider (2019) Automotive industry trends: IoT connected smart cars and vehicles. https://www.businessinsider.com/internet-of-things-connected-smart-cars-2016-10?r=US&IR=T

[30] Indu SK (2019) Internet of vehicles (IoV): evolution, architecture, security issues and trust aspects. Int J Recent Technol Eng 7(6):2019

[31] Bonomi F (2013) The smart and connected vehicle and the internet of things, synchronization. In: Telecommunication systems, pp 1–53

[32] Wan J, Zhang D, Zhao S, Yang L, Lloret J (2014) Context-aware vehicular cyber-physical systems with cloud support: architecture, challenges, and solutions. IEEE Commun Mag 52 (8):106–113

[33] Gandotra P, Kumar Jha R, Jain S (2017) A survey on device-to-device (D2D) communication: architecture and security issues. J Netw Comput Appl 78:9–29

[34] Aliyu A, Abdullah AH, Kaiwartya O, Cao Y, Lloret J, Aslam N, Joda U (2018) Towards video streaming in IoT environments: vehicular communication perspective. Comput Commun 118:93–119

[35] Alouache L, Nguyen N, Aliouatz M, Chelouah R (2018) Toward a hybrid SDN architecture for V2V communication in IoV environment. In: 2018 fifth international conference on software defined systems (SDS), Barcelona, Spain, https://doi.org/10.1109/sds.2018.8370428, 23–26 Apr 2018

[36] Raza S, Wang S, Ahmed M, Anwar MR (2019) A survey on vehicular edge computing architecture, applications, technical issues, and future directions. Wirel Commun Mob Comput
Sherazi HHR, Khan ZA, Iqbal R, Rizwan S, Imran MA, Awan K (2019) A heterogeneous IoV architecture for data forwarding in vehicle to infrastructure communication. https://doi.org/10.1155/2019/3101276.

Shi W, Zhou H, Li J, Xu W, Zhang N, Shen XSN (2018) Drone assisted vehicular networks: architecture, challenges and opportunities. IEEE Netw 99:1–8. https://doi.org/10.1109/mnet.2017.1700206.

Storck CR, Duarte-Figueiredo F (2019) 5G V2X ecosystem providing internet of vehicles. Special issue “Recent advances in software-defined internet of vehicles (SDIoV)”. Sensors 19(3):1–20. https://doi.org/10.3390/s19030550.

Wu C, Liu Z, Zhang D, Yoshinaga T, Ji Y (2018) Spatial intelligence towards trustworthy vehicular IoT. IEEE Commun Mag 56(10):22–27. https://doi.org/10.1109/mcom.2018.1800089.

Dey KC, Rayamajhi A, Chowdhury M, Bhavsar P, Martin J (2016) Vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication in a heterogeneous wireless network–performance evaluation. Transp Res Part C: Emerg Technol 68:168–184

Singh PK, Vij P, Vyas A, Nandi SK, Nandi S (2019) Elliptic curve cryptography based mechanism for secure wi-fi connectivity. In: International conference on distributed computing and internet technology, January 2019. Springer, Cham, pp 422–439

Naik G, Choudhury B, Park J (2019) IEEE 802.11bd & 5G NR V2X: evolution of radio access technologies for V2X communications. CoRR https://arxiv.org/abs/1903.08391.

Shin S, Forte AG, Rawat AS, Schulzrinne H (2004) Reducing MAC layer handoff latency in IEEE 802.11 wireless LANs. In: Proceedings of the second international workshop on mobility management & wireless access protocols, October 2004. ACM, New York, pp 19–26

Singh PK, Chattopadhyay S, Bhale P, Nandi S (2018) Fast and secure handoffs for v2i communication in smart city wi-fi deployment. In: International conference on distributed computing and internet technology, January 2018. Springer, Cham, pp 189–204

Dutta A, Famolari D, Das S, Ohba Y, Fajardo V, Tanfuchi K, Schulzrinne H (2008) Media-independent pre-authentication supporting secure inter domain handover optimization. IEEE Wirel Commun 15(2):55–64

Chekkouri AS, Ezzouhairi A, Pierre S (2015) Connected vehicles in an intelligent transport system. In: Vehicular communications and networks. Woodhead Publishing, Cambridge, pp 193–221

Haziza N, Kassab M, Knopp R, Härr J, Kaltenberger F, Agostini P, Aniss H (2013) Multi-technology vehicular cooperative system based on Software Defined Radio (SDR). In: International workshop on communication technologies for vehicles, May 2016. Springer, Berlin, pp 84–95

Lampropoulos G, Passas N, Kaloxyllos A, Merakos L (2007) A flexible UMTS/WLAN architecture for improved network performance. Wirel Pers Commun 43(3):889–906

Sadiq AS, Khan S, Ghafoor KZ, Guizani M, Mirjalili S (2018) Transmission power adaption scheme for improving IoV awareness exploiting: evaluation weighted matrix based on piggybacked information. Comput Netw 137:147–159

Sadiq AS, Bakar KA, Ghafoor KZ, Lloret J (2013) Intelligent vertical handover for heterogeneous wireless network. In: Proceedings of the world congress on engineering and computer science, vol 2, pp 774–779, October 2013

Al-Hader M, Rodzi A, Sharif AR, Ahmad N (2009) Smart city components architecture. In:2009 international conference on computational intelligence, modelling and simulation. IEEE, pp 93–97
[53] Kumar A, Prakash J, Dutt V (2014) Understanding human driving behavior through computational cognitive modeling. In: International conference on internet of vehicles. Springer, pp 56–65
[54] Atzori L, Iera A, Morabito G, Nitti M (2012) The social internet of things (SIOT)–when social networks meet the internet of things: concept, architecture and network characterization. Comput Netw 56(16):3594–3608
[55] Adly AS (2019) Technology trade-offs for IIoT systems and applications from a developing country perspective: case of Egypt. In: The internet of things in the industrial sector: security and device connectivity, smart environments, and industry 4.0. Computer communications, networks. Springer International Publishing, pp 299–319. https://doi.org/10.1007/978-3-030-24892-5_13
[56] Barba CT, Mateos MA, Soto PR, Mezher AM, Igartua MA (2012) Smart city for VANETs using warning messages, traffic statistics and intelligent traffic lights. In: 2012 IEEE intelligent vehicles symposium. IEEE, pp 902–907
[57] Kanter T, Rahmani R, Li Y, Xiao B (2014) Vehicular network enabling large-scale and real-time immersive participation. In: International conference on internet of vehicles. Springer, pp 66–75
[58] Balen J (2015) Spatio-temporal distributed background data storage and management system in VANETs. In: Proceeding of the 3rd GI/ITG KuVS Fachgespräch inter-vehicle communication (FG-IVC 2015), Ulm, Germany, vol 13(3), pp 22–25. https://bib.irb.hr/datoteka/787684.Balen-spatio-temp.pdf
[59] Förster D, Bosch R (2015) Discussing different levels of privacy protection in vehicular ad-hoc networks. In: Proceeding of the 3rd GI/ITG KuVS Fachgespräch inter-vehicle communication (FG-IVC 2015), Ulm, Germany, vol 13(3), pp 29–31
[60] Schmidt RK, Leinmüller T, Schoch E, Held A, Schäfer G (2008) Vehicle behavior analysis to enhance security in VANETs. In: Fourth workshop on vehicle to vehicle communications (V2VCOM 2008), Eindhoven, The Netherlands
[61] Wu C, Liu Z, Zhang D, Yoshinaga T, Ji Y (2018) Spatial intelligence towards trust worthy vehicular IoT. IEEE Commun Mag 56(10):22–27. https://doi.org/10.1109/mcom.2018.1800089