Enhancing Intrusion Detection in Vehicular Networks through Deep Learning Approaches

by

Kanika AGGARWAL

THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
IN PARTIAL FULFILLMENT OF A MASTER'S DEGREE
WITH THESIS IN ELECTRICAL ENGINEERING
M.A.Sc.

MONTREAL, "SEPTEMBER 16, 2023"

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

M. Georges Kaddoum, Thesis Supervisor
Department of Electrical Engineering, École de technologie supérieure

Ms. Diala Nabulsi, Chair, Board of Examiners
Department of Software and IT Engineering, École de technologie supérieure

Ms. Kuljeet Kaur, Member of the Jury
Department of Electrical Engineering, École de technologie supérieure

THIS THESIS WAS PRESENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC

ON "AUGUST 9, 2023"

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisor, Professor Georges Kaddoum, for his invaluable support, guidance, and patience throughout my M.A.Sc. study. His expertise and motivation have been instrumental in the successful completion of this dissertation and my academic journey.

I would also like to extend my thanks to the members of my jury for their time and effort in evaluating this dissertation.

I extend my heartfelt appreciation to Imene for being a constant source of encouragement and support throughout these years.

I would like to acknowledge and express my heartfelt thanks to my grandmother, parents, in-laws, younger brother, and all my family members for their endless support and encouragement. Their belief in my abilities has been a constant source of motivation.

Last but certainly not least, I want to express my deepest appreciation to my husband, Nitish. His unwavering support, encouragement, and presence have been my pillar of strength throughout this long journey. His constant cheerfulness and belief in me have been my pillars of strength throughout the good and challenging times.

I am truly fortunate to have such a remarkable support system, and I am grateful for each and every person who has played a role in my academic and personal growth.

Améliorer la détection des intrusions dans les réseaux de véhicules grâce à des approches d'apprentissage en profondeur

Kanika AGGARWAL

RÉSUMÉ

L'expansion récente de l'Internet des objets (IoT) a transformé les réseaux de véhicules en Internet des véhicules (IoV), où les véhicules modernes sont exposés à divers nouveaux types de cyberattaques. Afin de remédier à ces vulnérabilités, les systèmes de détection d'intrusion (IDS) jouent un rôle crucial dans la détection efficace des attaques avec une grande précision et un faible taux de fausses alarmes. Les approches traditionnelles d'apprentissage automatique ont été utilisées pour identifier les intrus dans le réseau, mais elles souffrent souvent d'une faible précision de détection et d'une grande complexité, ce qui les rend mal adaptées aux attaques dynamiques. Par conséquent, il existe un besoin pour un IDS avancé adapté aux scénarios en temps réel.

Pour améliorer la sécurité de l'IoV, nous proposons un nouveau système de détection d'intrusion basé sur un modèle d'apprentissage en profondeur (DL) hybride génératif. Le modèle proposé combine le auto-encodeur variationnel à mémoire longue et à court terme (LSTMVAE), les unités récurrentes bidirectionnelles fermées (BiGRU) et un classificateur softmax. Le LSTMVAE est utilisé comme technique d'extraction de caractéristiques statistiques capable d'apprendre des séries chronologiques et des données multivariées à partir du réseau IoV. Les caractéristiques extraites sont ensuite introduites dans le classificateur BiGRU et softmax pour l'identification et la classification des cyber-attaques potentielles dans le réseau IoV. Les résultats expérimentaux basés sur l'ensemble de données ToN-IoT valident les performances supérieures de l'IDS proposé par rapport aux techniques de base couramment utilisées.

En tirant parti des atouts du DL et des modèles génératifs, l'IDS proposé offre une solution plus efficace pour la détection des attaques dans les réseaux IoV. Il répond aux limites des approches traditionnelles d'apprentissage automatique et démontre une précision et des performances améliorées dans l'identification et la classification des cyberattaques. Cette recherche contribue à renforcer la sécurité des systèmes IoV et à atténuer les risques associés aux menaces émergentes.

Mots-clés: Système de détection d'intrusion, Internet des véhicules (IoV), Sécurité, Mémoire longue à court terme, Apprentissage en Profondeur

Enhancing Intrusion Detection in Vehicular Networks through Deep Learning Approaches

Kanika AGGARWAL

ABSTRACT

The recent expansion of the Internet of Things (IoT) has transformed vehicular networks into the Internet of Vehicles (IoV), where modern vehicles are exposed to various new types of cyber-attacks. In order to address these vulnerabilities, Intrusion Detection Systems (IDS) play a crucial role by effectively identifying and detecting attacks with a high level of accuracy while minimizing false alarms. Traditional Machine Learning (ML) approaches have been utilized to identify intruders in the network. However, they often suffer from low detection accuracy and high complexity, making them ill-suited for dynamic attacks. Therefore, there is a need for an advanced IDS suitable for real-time scenarios.

To enhance the security of IoV, we propose a novel IDS based on a generative hybrid deep learning (DL) model. The proposed model combines the Long Short-Term Memory Variational AutoEncoder (LSTMVAE), Bidirectional Gated Recurrent Units (BiGRU), and a softmax classifier. The LSTMVAE is employed as a statistical feature extraction technique capable of learning time series and multivariate data from the IoV network. The extracted features are then fed into the BiGRU and softmax classifier for the identification and classification of potential cyber-attacks in the IoV network. Experimental results based on the ToN-IoT dataset validate the superior performance of the proposed IDS over commonly used baseline techniques.

By leveraging the strengths of DL and generative models, the proposed IDS offers a more effective solution for attack detection in IoV networks. It addresses the limitations of traditional ML approaches and demonstrates improved accuracy and performance in identifying and classifying cyber-attacks. This research contributes to enhancing the security of IoV systems and mitigating the risks associated with the emerging threats.

Keywords: Intrusion Detection System, Internet of Vehicles (IoV), Security, Long Short-Term Memory, Deep Learning

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

LIST OF ALGORITHMS

Page

LIST OF ABBREVIATIONS

5G	Fifth Generation
AI	Artificial Intelligence
AE	AutoEncoder
AUC	Area under the ROC Curve
AVs	Autonomous Vehicles
BiGRU	Bidirectional Gated Recurrent Unit
CAN	Controller Area Network
CatBoost	Categorical Boosting
CM	Confusion Matrix
CNN	Convolutional Neural Network
D2D	Device-to-Device
DL	Deep Learning
DoS	Denial of Service
DDoS	Distributed Denial of Service
DNN	Deep Neural Network
DSRC	Dedicated Short Range Communications
DSAE	Deep Stacked Auto Encoder
DR	Detection Rate
ECUs	Electronic Control Units

FAR	False Alarm Rate
GRU	Gated Recurrent Unit
HL	Hidden Layer
HN	Hidden Node
IEEE	Institute of Electrical and Electronics Engineers
IDS	Intrusion Detection System
IoT	Internet of Things
IoV	Internet of Vehicles
LightGBM	Light Gradient Boosting Machine
LSTM	Long Short-Term Memory
LSTMVAE	Long Short-Term Memory Variational AutoEncoder
LTE	Long-Term Evolution
MEC	Multi-access Edge Computing
MITM	Man-in-the-Middle
ML	Machine Learning
NN	Neural Network
OBUs	On-Board Units
PCA	Principal Component Analysis
PR	Precision Rate
RC	Recall

RNNs	Recurrent Neural Networks
RFID	Radio Frequency Identification
ROC	Receiver Operating Characteristic
RSUs	Road Side Units
SAGS	Space-Air-Ground-Sea
SIEM	Security Information and Event Management
SHAP	Shapley Additive exPlanations
SVM	Support Vector Machine
TCCs	Traffic Command Centres
VANETs	Vehicular Ad-hoc Networks
V2H	Vehicle-to-Human
V2I	Vehicle-to-Infrastructure
V2N	Vehicle-to-Network
V2S	Vehicle-to-Sensor
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
WLAN	Wireless Local Area Network
XGBoost	Extreme Gradient Boosting
XSS	Cross-Site Scripting

LIST OF SYMBOLS AND UNITS OF MEASUREMENTS

σ	Activation function for the encoder
σ'	Activation function for the decoder
$\overrightarrow{G_T}$	Candidate cell state
$\tilde{\mathbb{C}}_T$	Cell input activation
$\mathbb{C}_T$	Cell state of LSTM
$\mathbb{F}_T$	Forget gate activation
h	LSTM units
$\overrightarrow{h_T}$	Final State of BiGRU
$\mathbb{I}_T$	Update activation
$\mathbb{K}$	Weight matrix for encoder
$\mathbb{K}'$	Weight matrix for decoder
$\overrightarrow{K}$	Weight matrix of BiGRU
$\mathbb{O}_T$	Output gate of LSTM
$\overrightarrow{R_T}$	Reset gate of BiGRU
$\mathbb{S}$	Bias for encoder
$\mathbb{S}'$	Bias for decoder
$\overrightarrow{\mathbb{S}}$	Bias Vector of BiGRU
$\mathbb{V}$	Weight matrix between the recurrent connections
$\mathbb{X}$	Dataset

$\overrightarrow{Z_T}$	Update gate of BiGRU
α	False Negative
β	True Negative
δ	True Positive
γ	False Positive
ψ	Decoder
ϕ	Encoder
$\oplus$	Elementwise summation for forward and backward vectors
$\odot$	Elementwise product
$\approx$	Latent representation
$\circ$	Relational composition between the previous and antecedent functions

INTRODUCTION

Over the past few decades, metropolitan cities worldwide have been grappling with the challenges posed by the increasing human population, which has resulted in a rapid growth of autonomous vehicles (AVs) on the roads. According to a recent report (Placek (2021)), it is projected that there will be approximately 54.2 million AVs by 2024. While AVs offer convenience, comfort, and improved safety for individuals, they also bring forth significant challenges in terms of environmental sustainability and energy conservation.

The proliferation of AVs has exacerbated environmental problems such as carbon emissions, urban congestion, pollution, fuel consumption, and traffic safety. The sheer volume of vehicles on the roads contributes to increased carbon emissions, leading to a negative impact on air quality and climate change. Additionally, the concentration of vehicles in urban areas often leads to traffic congestion, resulting in reduced mobility, longer travel times, and increased fuel consumption.

Integrating vehicular ad-hoc networks (VANETs) with the IoT has given rise to a new paradigm known as the IoV. IoV is a complex network system connecting users, vehicles, and smart devices through the Internet, leveraging communication and information technologies (Kaiwartya *et al.* (2016)). In an IoV system, users refer to the humans involved, including pedestrians, drivers, and passengers. They consume services provided by the network, such as auto-breaking, emergency calls, car surveillance, and lane change warnings. Vehicles within the IoV system act as nodes or smart objects equipped with wireless communication devices (e.g., cellular technology, wireless antennas) and advanced onboard sensors (e.g., Lidar, On-Board Units, Radar). These vehicles can communicate with each other and access resources such as cloud storage and computing (Kaiwartya *et al.* (2016)).

By leveraging the power of connectivity, data sharing, and advanced technologies, IoV has the potential to transform transportation systems, making them more efficient, intelligent, and

sustainable. The integration of AVs with IoV opens up opportunities for innovative services and solutions to address the challenges of urban congestion, pollution, and energy consumption, leading to a more connected and intelligent future of transportation.

Within the IoV infrastructure, AVs generate valuable and sensitive road data through the use of wireless communication devices and onboard sensors. This data can be classified into two categories: on-road data and on-board data (Sherazi, Iqbal, Ahmad, Khan & Chaudary (2019)). On-road data includes information related to events occurring on the road, such as traffic lights, blind spots, pilot camera videos, and inter-vehicle distances. On the other hand, on-board data refers to the data collected from sensors and systems within the vehicle itself, including engine parameters (such as temperature), velocity, fuel consumption, and braking information. By connecting and sharing this data among Vehicle-to-Everything (V2X) nodes, the IoV system can extend its perception capabilities, enhancing transportation efficiency, alleviating traffic congestion, and maximizing the existing road capacity.

The goal of IoV is to achieve coordinated development among AVs, the environment, and humans, resulting in an enhanced driving experience and a reduction in traffic issues through advanced navigation systems. In addition to safe driving, IoV aims to provide logistics, transportation, auto insurance, road infrastructure maintenance, and improve automobilism capacity and intelligence levels. Thus, it will provide users with intelligent, safe, comfortable, and efficient services.

Problem Statement

The interconnected nature of IoV, with its high dynamic topology and extensive internet connectivity, exposes AVs and the entire network to potential security risks. The dissemination of large amounts of information throughout the network creates opportunities for malicious activities such as data theft, tampering, eavesdropping, and malicious routing that can compromise the security of users and the overall network. These security threats not only endanger the lives

of users but also compromise the overall security of the network. The IoV network faces security challenges in four main areas (Man *et al.* (2021)): IoV communication, IoV cloud platforms, IoV mobile terminals, and vehicle security.

In addition to security concerns, safety is a crucial issue in the IoV. Hackers and intruders can exploit network communications to broadcast false or misleading information, take remote control of vehicles, or launch attacks that compromise the integrity, confidentiality, availability, authenticity, reliability, and privacy of the network, vehicles, and users. Real-world examples, such as hackers tricking Tesla's Autopilot software to change lanes into opposing traffic (Huddleston (2019)) or hijacking the digital systems of a Jeep Cherokee over the internet (Greenberg (2016)), highlight the need to prioritize safety, security, and privacy in IoV systems.

The most common cyber-attacks in the IoV (Li, Zuo, Song & Lv (2021)) include ransomware, backdoors, password, denial of service (DoS), man-in-the-middle (MITM), scanning, cross-site scripting (XSS), data injection, and distributed denial of service (DDoS). To address these threats, there is an urgent need for advanced IDS capable of efficiently detecting attacks with a high accuracy while minimizing false alarms.

Objective and Methodology

The main objective of this dissertation is to propose an effective solution that ensures privacy, trust, and security for smart vehicles. The proposed solution involves the development of a DL-based IDS to identify and classify potential cyber-attacks in the IoV network.

The proposed IDS introduces a generative hybrid DL model known as LSTMVAE-BiGRU, which combines LSTMVAE with BiGRU. The LSTMVAE performs encoding and decoding in the LSTMVAE paradigm, capturing the most important features of the training time series data by constraining the latent space dimensions to be smaller than the input. The BiGRU captures

local dependencies in a two-way time flow, considering both forward and backward directions for learning latent representations. The softmax layer is used for multi-class attack detection.

To achieve the primary goal, the dissertation follows a structured approach:

- **Theoretical Background:** The evolution of IoV networks and the motivation behind their development are described. The security issues involved in IoV networks are discussed. A detailed review of Artificial Intelligence (AI) based intrusion detection techniques in the literature and their associated challenges is presented. The dissertation also provides an in-depth discussion of neural networks and their types.
- **System Model:** A detailed description of the system model representing the proposed solution for identifying and classifying potential cyber-attacks in the IoV network is provided.
- **Hybrid DL Model:** An in-depth explanation of the hybrid DL model is given, which involves combining LSTMVAE, BiGRU, and a softmax classifier. The mathematical representations for each component of the proposed IDS architecture are described.
- **Dataset and Simulations:** The dataset used for the experiments is described, followed by simulations conducted to assess the efficacy of the proposed solution. A comparison is made against baseline techniques found in the literature that utilize supervised ML methods.
- **Performance Analysis:** The performance of the proposed model is analyzed using different performance metrics to assess its effectiveness in detecting and classifying cyber-attacks within the IoV network.

By following this structured approach, the dissertation aims to present a comprehensive solution to enhance the privacy, trust, and security in smart vehicles within the IoV network while effectively identifying and mitigating potential cyber-attacks.

Publications

The research presented in this dissertation has been accepted for publication in the proceedings of the IEEE Global Communications Conference (GLOBECOM 2023, Kuala Lumpur, Malaysia) under the title "LSTM-Based Hybrid Intrusion Detection System for Internet of Vehicles".

Dissertation Organization

The organization of this dissertation is structured into three chapters, each addressing specific aspects of the proposed solution. Here is a detailed breakdown of the chapters:

Chapter 1 provides a comprehensive review of relevant topics related to the dissertation. The topics covered include:

- **IoV network model and security approaches:** An overview of the IoV network model, its characteristics, and the security challenges it faces.
- **AI-based intrusion detection techniques:** A detailed exploration of intrusion detection techniques that utilize AI in securing IoV networks.
- **Theory and basic concepts of neural networks:** An in-depth explanation of the theory and fundamental concepts underlying neural networks, which form the basis of the proposed hybrid DL model.

Chapter 2 focuses on the proposed hybrid DL-based intrusion detection model. The contents include:

- **Introduction to the system model:** A detailed description of the system model that represents the proposed solution for identifying and classifying potential cyber-attacks in the IoV network.
- **A generative hybrid DL-based model:** A comprehensive examination of the model, which combines the LSTMVAE, the BiGRU, and a softmax classifier.

Finally, Chapter 3 focuses on the practical implementation and evaluation of the proposed solution. It includes:

- **Dataset description:** An explanation of the dataset used to conduct simulations and evaluate the proposed solution's performance.
- **Simulation methodology:** A detailed description of the methodology used to simulate the proposed solution and compare it against baseline techniques from the literature that employ supervised ML methods.
- **Performance metrics:** An overview of the performance metrics employed to analyze and evaluate the efficiency of the proposed model.
- **Results:** The simulation results are presented and compared with the baseline techniques, highlighting the advantages and effectiveness of the proposed solution.

Figure 0.1 illustrates a visual representation of this thesis contribution.

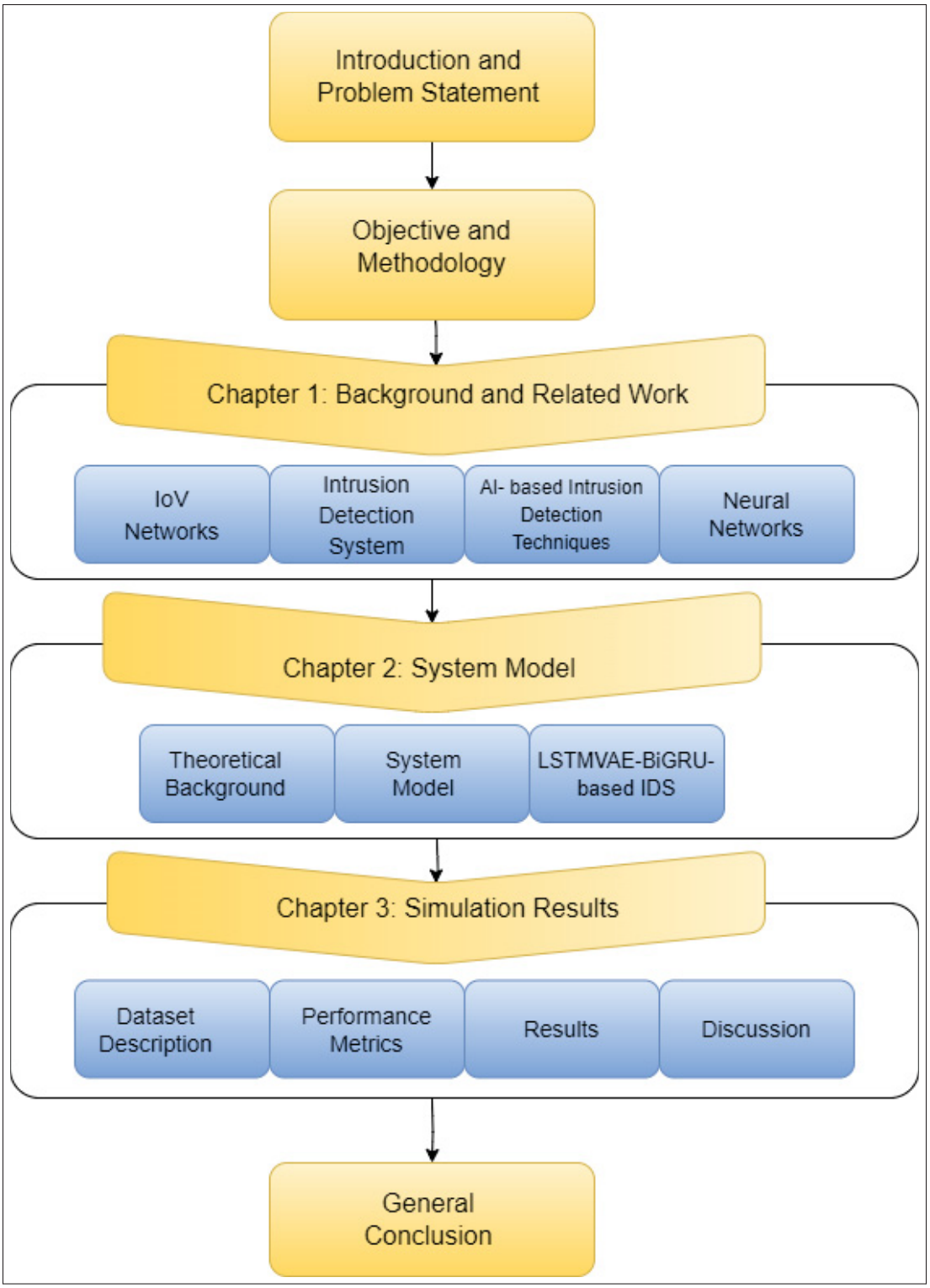

Figure 0.1 Chapters Diagram

CHAPTER 1

BACKGROUND AND RELATED WORK

1.1 IoV Networks

The IoV has emerged as a disruptive technology, extending the structure, scale, and applications of traditional VANETs. VANETs are designed to connect vehicles in an ad hoc manner, enabling real-time communication among smart vehicles and roadside units. However, VANETs have faced challenges in generating commercial interest due to various issues, such as lack of cloud connectivity, ad hoc architecture, limited reach, and limited computation ability. Consequently, VANETs have been unable to provide sustainable global services and applications to users, resulting in only a few developed countries like Japan and the USA having basic VANET implementations.

In contrast, IoV represents a large-scale network that integrates humans, smart vehicles, the surrounding environment, and other heterogeneous networks (Taslimasa *et al.* (2023)); (Kaiwartya *et al.* (2016)). It is an open and dynamic heterogeneous network characterized by higher controllability, manageability, credibility, and operationalization. Within the IoV infrastructure, modern vehicles are equipped with advanced onboard sensors and networking capabilities connected to the Internet. These sensors collect road data and enable communication and sharing with other vehicles, smart devices, and surrounding objects through V2X wireless communications. V2X technologies encompass vehicle-to-infrastructure (V2I), vehicle-to-human (V2H), vehicle-to-network (V2N), vehicle-to-sensor (V2S), and vehicle-to-vehicle (V2V) communications (Tang, Mao, Kato & Gui (2021)). These technologies have paved the way for enhanced driving experiences and intelligent transportation systems.

The IoV network possesses several special characteristics, as mentioned in the study by Magaia *et al.* (2020):

- **Dynamic Topology:** Due to the high mobility of vehicles in the IoV network, the network topology undergoes frequent changes. The movement of vehicles results in dynamic and unpredictable connectivity patterns within the network.
- **Variable Network Density:** The density of the IoV network fluctuates depending on the quantity of traffic nodes present in a particular area. During peak traffic hours, the network density can be very high, while it may be lower during normal hours.
- **Predictable Mobility:** The mobility of nodes in the IoV network follows predictable patterns due to the constraints imposed by the road layout, network topology, traffic lights, and road signs. Additionally, nodes communicate with nearby moving nodes, contributing to better predictability in terms of node mobility.
- **Non-uniform Distribution of Nodes:** The distribution of nodes in the IoV network is influenced by various factors, including geographical location and road network topology. As a result, the network's connectivity can vary across different areas, leading to a non-uniform distribution of nodes.

In addition to the special characteristics of the IoV network, several factors contribute to the complexity of the network:

- **External and Internal Sensors:** Vehicles in the IoV network are equipped with both external and internal sensors. External sensors, such as parking sensors and cameras, are installed outside the vehicle to collect real-time information about the vehicle's surroundings, including obstacles and other vehicles. Internal sensors provide data about the vehicle's internal conditions, like temperature, localization, and braking system status. These sensors include light detection and ranging, tire-pressure monitoring systems, and automotive sensors for monitoring fuel levels and brakes.
- **Driver's Social Profile:** The driver's social profile encompasses their messages, pictures, tweets, and other information that can help characterize the driver's state of mind. Analyzing this data can provide insights into driver preferences, emotions, and potential distractions. Incorporating the driver's social profile into the IoV network adds another layer of complexity in terms of data collection, analysis, and privacy considerations.

- **Beacon Signals/Messages:** Beacon messages are responsible for conveying various conditions, such as possible delays on the route or the drivability status of the vehicle. Processing and interpreting these beacon signals effectively is crucial for ensuring efficient and reliable communication within the IoV network.

1.1.1 IoV Network Model

The general hierarchical IoV architecture (Kaiwartya *et al.* (2016)); (Magaia *et al.* (2020)) comprises three main layers: the perception layer, the network layer, and the application layer. The architecture is illustrated in Fig. 1.1.

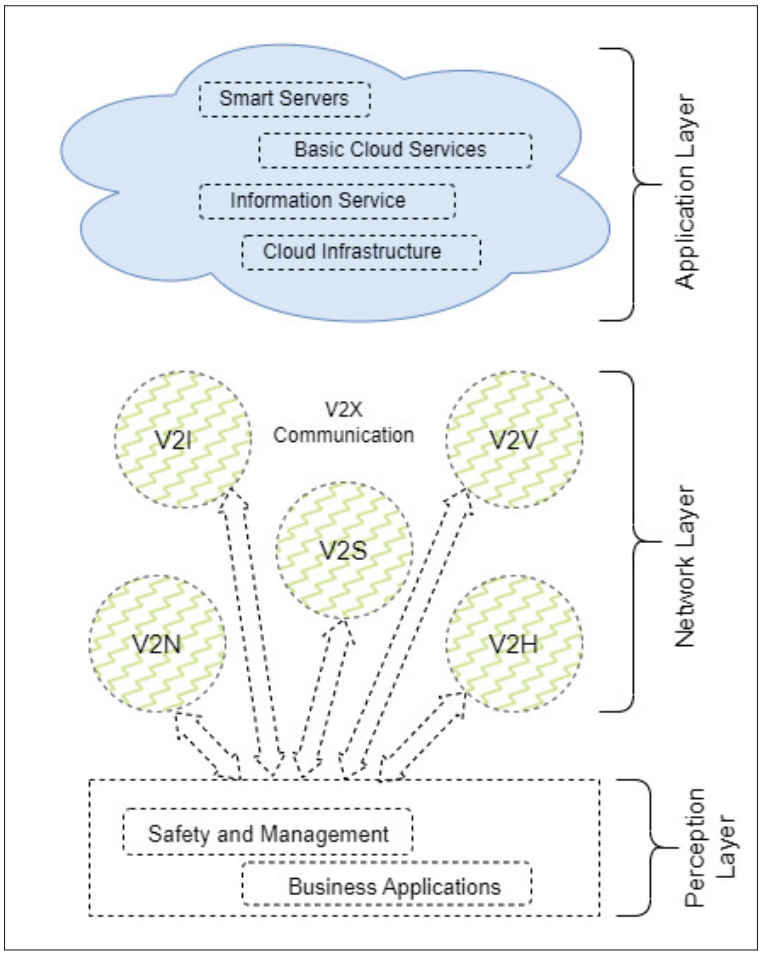

Figure 1.1 IoV three-layer Architecture

Each layer has specific functions and contributes to the overall functioning of the IoV system. Here is a brief description of each layer:

1.1.1.1 Perception Layer

The perception layer, also known as the client layer, consists of sensors that collect environmental information and detect events and situations. This layer consists of external sensors installed on the vehicle (e.g., cameras, radars) as well as internal sensors (e.g., temperature sensors, GPS receivers). The sensors monitor the vehicle's surroundings, capture driving patterns, and gather data related to road conditions, traffic, and other relevant factors. Additionally, this layer incorporates technologies like radio frequency identification (RFID) tags for object tracking and perception of the surrounding environment. The collected data is then processed and transmitted to higher layers for further analysis and decision-making.

1.1.1.2 Network Layer

The network layer is responsible for facilitating communication among vehicular nodes, as well as other entities and networks involved in the IoV network. It enables different types of wireless communication technologies, such as Bluetooth, Wireless Local Area Network (WLAN), Wi-Fi, or fifth-generation (5G) mobile communications networks. The network layer ensures the interconnectivity of various communication modes within the IoV network, including V2I, V2N, V2V, V2S, and V2H interactions.

1.1.1.3 Application Layer

The application layer is responsible for storage, processing, analysis, and decision-making based on the collected data from the perception layer. It encompasses a range of applications and services that leverage the IoV network's capabilities. Some key functions of the application layer include ensuring road safety, enhancing traffic efficiency, supporting autonomous driving features, and providing infotainment services to vehicle occupants. The application layer

processes the data collected from the perception layer, applies various algorithms and techniques for analysis, and generates actionable insights and responses to different scenarios.

By having these three layers in the hierarchical IoV architecture, the system can handle scalability, heterogeneity, and the specific requirements of the IoV paradigm effectively. Each layer performs its designated tasks, enabling seamless communication, data processing, and intelligent decision-making within the IoV network.

1.1.2 IoV Security

IoV is critical in supporting various applications such as traffic and parking management, tracking systems, and transport efficiency. The projected growth of AVs further emphasizes the importance of IoV. According to Statista (Placek (Aug.2021)), the United States is expected to have approximately 146 million AVs by 2030. However, the increased number of V2I and V2V communication links resulting from the proliferation of AVs presents a significant challenge in terms of securing the IoV network.

The interconnected nature of the IoV network, which encompasses smart vehicles, surrounding environments, and public internet connections, exposes it to various cyber-attacks. The presence of these attacks represents a major threat to the trust, privacy, and security of modern vehicles, potentially leading to severe risks to human lives. Furthermore, intrusions in IoV systems target the integrity, confidentiality, availability, authenticity, reliability, and privacy of both the network and the vehicles themselves. Thus, the security of IoV is a major concern within vehicular networks. Security threats in IoV can be broadly categorized into two types (Khraisat, Gondal, Vamplew & Kamruzzaman (2019)): Inter-Vehicle attacks and Intra-Vehicle attacks.

1.1.2.1 Inter-Vehicle Attacks

Inter-Vehicle communication in the IoV network involves the exchange of information between smart vehicles and various other entities present on the road, including smart devices, road infrastructures, other vehicles, and pedestrians. This communication aims to enhance the

driving experience and ensure safer interactions on the road. However, the establishment of communication channels, such as global systems for mobile communication and protocols like DSRC (Dedicated Short-Range Communication) and LTE (Long-Term Evolution), exposes smart vehicles to various cyber-attacks.

Cyber attackers can exploit vulnerabilities in the communication system to compromise the integrity, confidentiality, and availability of information exchanged between vehicles. They may attempt to gain control over the network by modifying information, disseminating false data, disrupting the network through the transmission of bulk data, or introducing malicious activities. Several common types of cyber-attacks can target inter-vehicle communication in the IoV network (Li *et al.* (2021)). These include:

- **Eavesdropping:** Attackers intercept and listen to communication between vehicles, potentially gaining access to sensitive information.
- **DDoS:** Attackers overwhelm the communication channels by flooding them with a massive volume of data, rendering the network unavailable to legitimate users.
- **MITM:** Attackers intercept and alter communication between vehicles, allowing them to manipulate or modify information exchanged between the parties.
- **Impersonation Attack:** Attackers masquerade as legitimate entities within the IoV network, deceiving vehicles and other entities into believing they are trustworthy.
- **Routing Attack:** Attackers manipulate routing information within the network, diverting communication between vehicles to unauthorized paths or disrupting the network's normal operation.
- **Phishing:** Attackers attempt to deceive vehicle users or other network entities by sending fraudulent messages or creating fake websites, aiming to trick them into revealing sensitive information or performing malicious actions.

1.1.2.2 Intra-Vehicle Attacks

Intra-Vehicle communication in the IoV network refers to the communication between Electronic Control Units (ECUs) and sensors within a vehicle. The Controller Area Network (CAN) bus is

a widely used protocol in the automotive industry for intra-vehicle communication. It offers real-time communication, low cost, a serial mechanism, and ease of installation (Taslimasa *et al.* (2023)).

However, the CAN bus has certain vulnerabilities that make it susceptible to attacks. One of the main vulnerabilities is the lack of an encryption protocol, which means that data transmitted on the bus can be easily intercepted and manipulated by attackers. Additionally, the CAN bus lacks an authentication mechanism, allowing intruders to insert malicious information into the intra-vehicle network and potentially gain control over the bus (De Araujo-Filho, Pinheiro, Kaddoum, Campelo & Soares (2021)). Some common attack types on the CAN bus (Li *et al.* (2021)) include:

- **Spoofing Attack:** Attackers send messages with forged source addresses to deceive the receiving ECUs. This can lead to unauthorized access or manipulation of critical vehicle functions.
- **Data Falsifying Attack:** Attackers modify the data being transmitted on the CAN bus, leading to false sensor readings or control commands, potentially affecting the vehicle's performance or safety.
- **DoS Attack:** Attackers flood the CAN bus with a high volume of messages, disrupting normal communication and rendering the bus unavailable for legitimate transmissions.
- **Fuzzing Attack:** Attackers inject random or unexpected data into the CAN bus to provoke software or system errors, potentially leading to system crashes or vulnerabilities being exposed.

Hence, these attacks highlight the importance of robust security measures in the IoV network to protect against unauthorized access, data manipulation, privacy breaches, and service disruptions.

1.2 Intrusion Detection System

IDS plays a crucial role in identifying potential cyber-security attacks that pose risks to the integrity, confidentiality, availability, authenticity, reliability, and privacy of the network (Oseni

et al. (2022)). It is a powerful tool, device, or software application that analyzes network traffic patterns to detect malicious activities.

The primary responsibility of an IDS is to continuously monitor the system or network for any suspicious activities or policy violations. When unauthorized behavior or violations are detected, the IDS typically alerts the system administrator or records the event centrally using a Security Information and Event Management (SIEM) system (He, Kim & Asghar (2023)). A SIEM system aggregates and analyzes relevant data from various sources across the entire network infrastructure, providing valuable insights to security teams for taking appropriate actions against security threats. IDS can be categorized into two main types: Signature-based IDS and Anomaly-based IDS.

1.2.1 Signature-based Intrusion Detection System

Signature-based IDS relies on analyzing network packets to identify specific patterns or sequences that match known malicious patterns or byte sequences, which are referred to as attack signatures. These signatures can be found in a series of packets, specific data sequences, or even in source and destination network addresses. To detect known attacks, a signature-based IDS maintains a database of attack signatures. Incoming network packets are compared against the signatures in the database, and if a match is found, the IDS triggers an alert or takes appropriate action. Signature-based approaches are generally simple and efficient, capable of operating in real time. They are particularly effective in identifying known attacks for which signatures have been defined.

However, one limitation of signature-based IDS is their inability to detect unknown and zero-day attacks (Khraisat *et al.* (2019)). Since these attacks do not have pre-defined signatures in the database, they can go undetected by the IDS. Therefore, while signature-based IDS are effective in detecting known attacks, they may not provide sufficient protection against emerging or previously unseen attack techniques.

1.2.2 Anomaly-based Intrusion Detection System

Anomaly-based IDS is designed to detect unknown or new attacks, particularly in response to the increasing number of malware threats. This type of IDS utilizes ML, knowledge-based, or statistical-based methods to create a model of normal or trustworthy network behavior. It then compares the observed behavior against this model and flags any deviations as potential anomalies or attacks. Unlike signature-based IDS that focuses on specific attack patterns, anomaly-based IDS analyzes behaviors associated with attacks, thereby increasing the chances of detecting, identifying, and mitigating malicious activity before it compromises the system.

Unlike signature-based IDS, ML-based anomaly detection methods have a more generalized property since the models can be trained according to specific applications and hardware configurations. The anomaly-based approach is capable of detecting abnormal behavior and identifying unknown and zero-day attacks. However, it may be prone to false positives, as legitimate activities that were previously unknown or unusual may trigger alerts as malicious. Another challenge faced by existing IDS is the potential performance degradation due to the time required for the detection process. Additionally, an effective operating algorithm is necessary to accurately analyze incoming packets and determine if they exhibit anomalous behavior. Efficient algorithms are required to accurately analyze incoming packets and minimize delays in the detection and response to potential threats.

1.3 Artificial Intelligence-based Intrusion Detection Techniques

Ensuring the security and safety of pedestrians and AVs within the IoV environment is paramount. To address the security challenges in vehicular networks, extensive research has been conducted in recent years. Many studies in the literature have explored the use of AI techniques (Ahmad, Shahid Khan, Wai Shiang, Abdullah & Ahmad (2021)), specifically ML methods (Talpur & Gurusamy (2021)) and DL models (Boualouache & Engel (2023)), for IDS in the context of vehicular networks. These works aim to develop robust and effective security mechanisms to safeguard the IoV environment.

1.3.1 Machine Learning-based IDS for IoV

Researchers have explored various ML-based techniques (Hachimi, Kaddoum, Gagnon & Illy (2020)) to enhance the security of vehicular networks. For example, in a study by Anzer & Elhadef (2018), a supervised learning-based IDS was developed to identify intruders in the IoV network based on V2V communications. However, their model was trained on the KDD Cup 1999 dataset, which may not be suitable for vehicular networks. Rani *et al.* (2023) presented a deep hierarchical ML-based IDS for Device-to-Device (D2D) communications. Sherazi *et al.* (2019) proposed a fuzzy logic-based framework implemented on a 6BR device to detect and prevent DDoS attacks in IoV communications. The performance of their system was evaluated using metrics such as response time, energy consumption, throughput, and average buffer usage. de Araujo-Filho *et al.* (2020) proposed a novel unsupervised-based IDS framework to identify different cyber-attacks within cyber-physical systems.

Sharma & Liu (2020) focused on misbehavior detection in IoVs using supervised ML approaches integrated with plausibility checks. Their model was compared with various supervised ML algorithms and plausibility checks for effectiveness. Yang, Shami, Stevens & De Rusett (2022) introduced an ensemble-based IDS model for IoV networks using ML techniques. They integrated three advanced ML algorithms (Extreme Gradient Boosting (XGBoost), Light Gradient Boosting Machine, and Categorical Boosting) and identified the best-performing model for each type of attack. Similarly, an IDS framework was implemented using different ML techniques. In this study, Kasongo (2023) applied an XGBoost-based feature selection algorithm. Addressing malicious security failures within software-defined networks (Miranda, Kaddoum, Bou-Harb, Garg & Kaur (2020)); (Miranda, Kaddoum, Boukhtouta, Madi & Alameddine (2022)), several intrusion detection schemes have been put forth. For example, Anbalagan *et al.* (2021) proposed a distributed ML scheme for software-defined IoV networks, considering the deployment of Road Side Units (RSUs) to improve transmission efficiency.

Yang, Moubayed, Hamieh & Shami (2019) proposed a tree-based IDS using ML algorithms to detect threats both on controller area network (CAN) bus and external networks. The proposed

model outperforms the existing methods by achieving a 2-3% increase in accuracy, detection rates, F1 scores, and a reduction in false alarms. De Araujo-Filho *et al.* (2021) introduced a novel ML intrusion prevention model to detect cyber-attacks in CANs. Pradhan, Mohanty & Seemona (2022) implemented an IDS model based on various ML algorithms (K-Nearest Neighbors, Decision Tree, Naive Bayes, Logistic Regression, Support Vector Machine (SVM), and Random Forest), and an ensemble model. Illy, Kaddoum, Moreira, Kaur & Garg (2019) introduced an ensemble-based anomaly detection method, in which various learning models, such as random forest are utilized to create an ensemble model. The proposed model is trained on several realistic datasets. Yang, Moubayed & Shami (2021) presented a multitiered hybrid IDS to detect known and unknown cyber-attacks on both external-vehicular and intravehicle networks using multiple ML algorithms.

However, many of these solutions exhibit drawbacks such as low detection accuracy, high complexity, and limited generalization capabilities. These limitations make them less suitable for real-time scenarios with dynamic attacks. Considering the complexity and connectivity of IoV networks, as well as the resource constraints, novel IDS mechanisms are needed to address the specific challenges posed by vehicular networks in real-time environments.

1.3.2 Deep Learning-based IDS for IoV

Owing to the explosion of vehicular traffic data, the DL models (Garg, Kaur, Kaddoum, Garigipati & Aujla (2021)) have garnered considerable interest from academia and the telecom industry. These are particularly appealing as they have the potential to effectively deal with heterogeneous, unstructured, and large volumes of data. Several research studies have explored the application of DL models in IDS for vehicular networks. For instance, the work in (Alladi, Kohli, Chamola, Yu & Guizani (2021)) deployed the detection engines on multi-access edge computing (MEC) servers in IoV networks. A DL-based IDS framework is proposed, which comprises DL engines to identify and classify malicious traffic. However, the proposed approach has considered fewer cyber-security attacks for simulation. Oseni *et al.* (2023) employed a Shapley Additive exPlanations (SHAP) scheme to study the output of a DL-based IDS in IoV

systems. After having insights into local and global explanations, the system would be more resilient to cyber-attacks. However, the considered approach is computationally expensive and vulnerable to adversarial attacks.

A convolutional neural network (CNN)-based IDS is proposed by Nie *et al.* (2020) to detect different intrusion attacks that aim at RSUs. The evaluation results showed that the proposed method achieves a high threat detection rate and low false rate as compared with principal component analysis (PCA), traditional shallow Neural Network (NN), and SVM methods using the same dataset. However, a limited set of attacks were considered in their work. Garg *et al.* (2019) designed a hybrid DL approach that combines CNN and grey wolf optimization for anomaly detection within cloud environments. Another approach in (Ahmed, Jeon & Ahmad (2021)), where the authors tried to solve the anomaly detection problem in the CAN bus by leveraging a CNN, specifically the VGG-16 architecture. The model is trained using the CAN-intrusion dataset, which includes fuzzy attacks, DoS attacks, and normal attacks.

An Intelligent IDS framework has been proposed in (Anbalagan, Raja, Gurumoorthy, Suresh & Dev (2023)) for a 5G V2X environment to efficiently broadcast messages regarding malicious AVs. A modified CNN architecture was employed to enhance the intrusion detection and classification of cyber-attacks. Almutlaq, Derhab, Hassan & Kaur (2022) used the rule extraction method from deep neural networks (DNNs) and implemented a two-stage IDS for intelligent transportation systems. The proposed model was evaluated using the car hacking dataset and four other traditional IDS datasets that represent intra-vehicle and external network communications, respectively. Illy, Kaddoum, de Araujo-Filho, Kaur & Garg (2022) proposed a hybrid DNN-based IDS framework in which multiple DNN models collaboratively undergo training.

Although different DL approaches have been proposed in the literature, most yield low detection accuracy and high false alarm rates. These techniques also face key challenges. Due to the fast movement of modern vehicles, the topology of IoV changes frequently, and thus, they access the network randomly. On the other hand, for intrusion detection, DL-based systems frequently require a large amount of network data to discover abnormalities. However, getting

such an enormous amount of data is a challenging issue. In addition, extracting the features of all possible patterns available in the IoV network is a challenging task since these features constitute the model's overall performance.

To address the challenges, recent studies have proposed LSTM architecture as one of the promising solutions for IDS. LSTMs are a special kind of Recurrent Neural Networks (RNNs) that can handle long-term dependencies (Tang *et al.* (2021)). The key aspect of LSTM-based RNNs is to remember the information for long periods, making them well-suited for conducting experiments on spatial data that exhibits temporal variations. For instance, Ashraf *et al.* (2020) presented a LSTM autoencoder-based intrusion detection framework that can identify abnormal behaviour in both external networks and CANs. Alferaidi *et al.* (2022) proposed a distributed DL model that combines CNN and extended LSTM for intrusion detection in IoV networks. Another DL-based threat intelligence scheme has been presented in (Al-Hawawreh, Moustafa, Garg & Hossain (2020)) to detect cyber-attacks from space-air-ground-sea (SAGS) networks. A deep stacked autoencoder (DSAE) was used to extract the hidden patterns of IoT-SAGS network traffic, and then, a Gated Recurrent Neural Network was applied to detect the cyber-attack types. However, these approaches are designed to model the univariate sequence representations. Thus, we propose a hybrid DL model to learn the time series and multivariate data from IoV networks.

To achieve the objective, the primary contributions of this research are summarized as follows:

- A novel IDS based on a generative hybrid DL model that combines LSTM Variational AutoEncoder with BiGRU is proposed.
- Specifically, a statistical feature extraction technique based on LSTMVAE is designed. The proposed LSTMVAE can learn time series and multivariate data from the IoV network efficiently. Second, the extracted features are used by the BiGRU and softmax classifier to differentiate between reliable and doubtful occurrences in the IoV network.
- The performance of the proposed IDS is evaluated using various evaluation metrics, and further compared with commonly used baseline techniques and recent state-of-the-art methods.

In the following section, we provide an in-depth review of the neural networks and their various types that have been implemented in our proposed solution.

1.4 Neural Networks

A neural network is a computational model structured like the human brain. It consists of interconnected artificial neurons or nodes arranged in a layered structure. The basic neural network architecture includes an input layer, one or more hidden layers, and an output layer. Each neuron in the network receives input from neurons in the previous layer, performs computations, and passes the output to neurons in the next layer. Deep neural networks consist of multiple hidden layers, allowing for more complex and hierarchical representations of data. In what follows, we will go through different types of neural networks.

1.4.1 Recurrent Neural Network

RNNs are capable of handling sequential data by taking into account not only the current input but also the previous outputs, as shown in Figure 1.2. All inputs and outputs in traditional neural networks are independent of each other. However, in some cases, a previous input is required. For instance, to predict the next word of a sentence, it is better to know which words came before it. RNNs addressed this challenge by incorporating hidden layers that aid in retaining information from previous inputs.

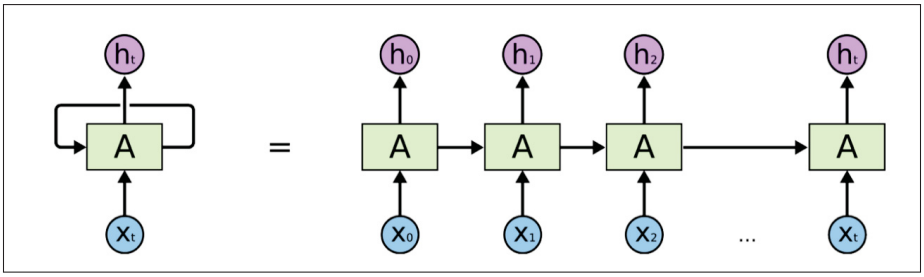

Figure 1.2 An unfolded Recurrent Neural Network
Taken from Olah (2015)

RNNs are useful in numerous applications (Boualouache & Engel (2023)), such as machine translation, image captioning, time series prediction, speech recognition, optimization, etc. However, the exploding gradient problem and vanishing gradient problem are two issues commonly associated with RNNs. These problems arise from the nature of backpropagation, the algorithm used to train neural networks. During backpropagation, gradients are calculated and propagated back through time, allowing for the updating of network weights. However, when the network has many layers or long sequences, the gradients can grow exponentially (exploding gradients) or diminish exponentially (vanishing gradients).

1.4.2 Long Short-Term Memory Network

LSTM networks are a type of RNNs that are specifically designed to address the vanishing gradient problem and handle long-term dependencies in sequential data. Unlike traditional RNNs, LSTMs have a more complex internal structure that allows them to capture and remember information over long periods of time.

In traditional RNNs, the repeating module consists of a single layer. In contrast, in LSTM networks, this module is replaced by a memory unit consisting of four interacting layers, as shown in Figure 1.3. These layers are the input gate, output gate, forget gate, and memory cell. This gated structure allows LSTM networks to control the flow of information through time and retain important information while discarding irrelevant or outdated information. In addition, the combination of these gates and memory cells allows LSTM networks to learn and remember time series with long time lags, even when the size of the lag is unknown. This makes them particularly effective in tasks (Boualouache & Engel (2023)) such as speech recognition, natural language processing, forecasting, etc. In what follows, we provide a detailed explanation of the functioning of LSTM networks.

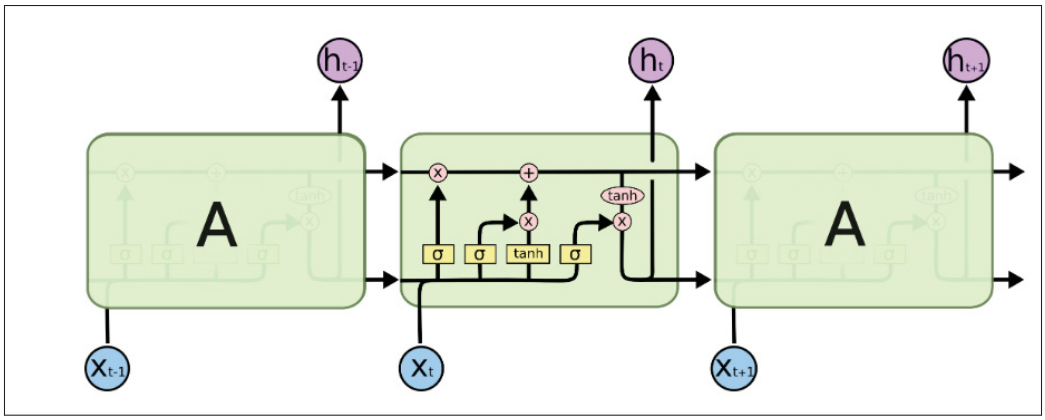

Figure 1.3 LSTM with four interacting layers
Taken from Olah (2015)

Step 1: Determine the amount of past information to remember and forget:

In LSTM, the first step is to evaluate the relevance of the past information stored in the memory cell. A sigmoid function is used for this decision, also known as forget gate. It decides how much of the past information should be retained or forgotten from the cell state. W and b are the linear weights and offsets associated with the gates, respectively. The forget gate looks at the output of the previous cell h_{t-1} and the current input x_t to output a number between 0 and 1 (Wang, Wang, Jiang, Xu & Wang (2023)). A '1' means to keep this entire piece of information, while a '0' value represents completely getting rid of the information.

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f). \tag{1.1}$$

Step 2: Decide which new information should be added to the current state:

This step has two parts, namely a sigmoid function and a *tanh* function. A sigmoid function decides which values will be updated while the *tanh* function creates a vector consisting of all possible values that can be added to the state and output the values from -1 to 1. Then, the cell state is updated by combining these two steps (Wang *et al.* (2023)):

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i)]. \tag{1.2}$$

$$\tilde{C_t} = tanh(Wc.[h_{t-1}, x_t] + b_C)]. \tag{1.3}$$

Step 3: Determine the output from the current cell state:

To determine the output of an LSTM, the first step is to apply a sigmoid layer. This layer is responsible for deciding which portion of the current cell state should be included in the final output. Then, we multiply the output of a sigmoid function with the cell state having $tanh$ function, where o_t, h_t, and C_t are the output gate activation, forget gate activation, and cell state activation, respectively. Finally, we reach the output of the memory cells (Wang *et al.* (2023)).

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o). \tag{1.4}$$

$$h_t = o_t * tanh(C_t). \tag{1.5}$$

1.4.3 Gated Recurrent Unit

Gated Recurrent Unit (GRU) is the modified version of RNNs that aims to effectively solve the vanishing gradient problem during training (Kostadinov (2017)). In general, the LSTM and GRU are designed similarly, except for the combination of forget gate and input gate into a single gate known as the update gate in GRU. In some cases, these both provide equally good performances. However, as compared to LSTM, GRU is less complex. Moreover, GRU has less number of parameters which helps in saving much time when training the neural network with a large amount of data. Also, it helps in improving the training speed and converges easily. In addition, the GRU can be trained to keep the information for longer periods of time without washing or removing information that seems irrelevant to the prediction. GRU networks make the training significantly more efficient by avoiding the risk of over-fitting.

1.4.4 Autoencoder

An autoencoder (AE) is a special type of feedforward neural network designed to reduce the dimensionality of the input data. It is designed in the 1980s by Geoffrey Hinton to solve unsupervised learning problems (Park *et al.* (2023)). AE is an unsupervised learning model that reconstructs the original input to the output by passing it through the tiny middle layer known as the bottleneck layer. Specifically, an AE consists of two networks: an encoder and a decoder. The encoder encodes/compresses the raw input data x into the encoded space of representation or latent code

$$y = f(xw + b), \tag{1.6}$$

where f denote the activation function of the encoder, and b and w denote the bias vector and the weight matrix, respectively. On the other hand, the decoder tries to decodes/decompresses the latent code and recreates the representation y into the corresponding input data $\bar{x}$

$$\bar{x} = f_1(yw' + b'), \tag{1.7}$$

where f_1 denote the activation function of the decoder, and b' and w' denote the bias vector and the weight matrix, respectively. Thus, an AE is trained to minimize the reconstruction error L

$$\begin{aligned} L(x, \bar{x}; w, w') &= \|x - \bar{x}\|_2^2 \\ &= \|x - f_1(w'.f(xw + b) + b')\| . \end{aligned} \tag{1.8}$$

1.5 Conclusion

The chapter started by providing a detailed review of IoV networks, including their hierarchical architecture and the challenges related to security, privacy, and safety. It emphasized the importance of developing effective IDS to address these issues. Several studies conducted by researchers were discussed, focusing on the utilization of ML and DL techniques for enhancing

the security of vehicular networks. In the latter part of the chapter, the focus shifted to neural networks and their relevance in identifying and mitigating attacks in IoV networks.

CHAPTER 2

SYSTEM MODEL

2.1 Introduction

In this chapter, we will provide a detailed description of the network model developed for our proposed IDS. We will begin by introducing the theoretical background; then we will provide an overview of the simulated environment and outline the key aspects. Furthermore, we will present a detailed explanation of the DL model employed in our system, which is responsible for feature extraction and multi-class attack classification.

In addition, we will present our proposed IDS solution as an algorithm. This algorithm will provide a step-by-step explanation of the entire process. Finally, we will present the mathematical representations of three main components of our system model: the Long Short-Term Memory Variational AutoEncoder, the Bidirectional Gated Recurrent Units, and the softmax classifier.

2.2 Network model of proposed IDS

The network model for the proposed IDS is shown in Fig. 2.1. The network model contains three entities: Traffic Command Centres (TCCs) located at cloud servers, RSUs, and OBUs. Typically, in the IoV network, vehicles are equipped with OBU units that record traffic and driving data, and share this information with RSUs or other nearby vehicles in the network. The RSUs are positioned at different geographical locations and share the vehicles' real-time information with TCCs. An attacker can easily target the OBUs located on the top of vehicles and can further attempt to exploit the telematics services of the vehicles. Since most vehicles have enough memory and computer power, we recommend installing the proposed IDS as a firmware or software within the OBUs.

Specifically, the proposed IDS is a generative hybrid DL model that combines LSTMVAE with BiGRU to learn time series and multivariate data from the IoV network, and is capable of differentiating between reliable and doubtful instances based on the most representative

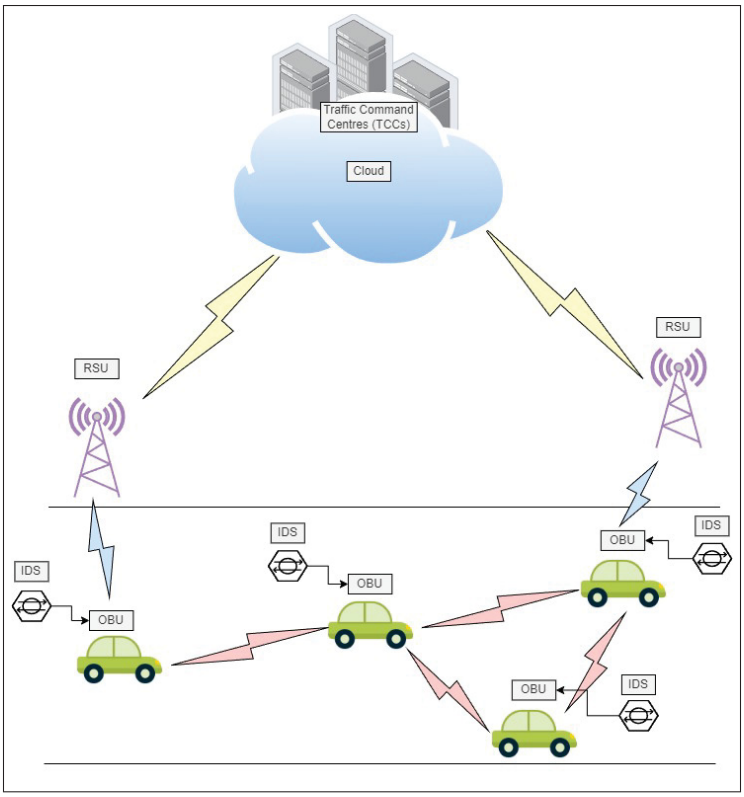

Figure 2.1 Network model of proposed IDS

attributes. The LSTMVAE approach is designed to ensure feature extraction, and BiGRU with softmax classifier is combined for attack classification. In the following section, we will discuss the proposed architecture in detail.

2.3 The LSTMVAE-BiGRU-based IDS

In the proposed LSTMVAE-BiGRU-based IDS, there are three main parts: LSTMVAE, BiGRU, and the softmax classifier.

2.3.1 The LSTMVAE-BiGRU Algorithm

In the LSTMVAE-BiGRU-based IDS architecture, the LSTM performs the encoding and decoding in the LSTMVAE paradigm because it combines the LSTM network with the AE. Furthermore, by limiting the latent space to have a dimension that is less than the input, the

Algorithm 2.1 Proposed LSTMVAE-BiGRU

procedure INPUT:(Read the Dataset)
  OUTPUT: Attack classification
  Pre-process Dataset
    Imputation of missing values
    Convert categorical features into numeric data
    Normalize data values between 0 and 1
  Divide Data into Training, Validation, and Testing sets.
  Feature Extraction using the LSTMVAE technique
    Perform encoding and decoding
    $\phi : \mathbb{X} \rightarrow \mathbb{K} := \approx = \sigma(\mathbb{K}\mathbb{X} + \mathbb{S})$
    $\psi : \mathbb{K} \rightarrow \hat{\mathbb{X}} = \sigma' (\mathbb{K}'\approx + \mathbb{S}')$
    Replace the latent features with
    $\approx_t = \mathbb{O}_T \circ \sigma_h (\mathbb{C}_T)$
    Add LSTM layers and perform encoding
    $\mathbb{C}_T = (\mathbb{F}_T \circ \mathbb{C}_{T-1}) + \left(\mathbb{I}_T \circ \tilde{\mathbb{C}}_{T-1}\right)$
    $\mathbb{F}_T = \sigma_G (\mathbb{K}_F \mathbb{X}_T + \mathbb{V}_F \approx_{T-1} + \mathbb{S}_F)$
    $\mathbb{I}_T = \sigma_G (\mathbb{K}_I \mathbb{X}_T + \mathbb{V}_I \approx_{T-1} + \mathbb{S}_I)$
    $\mathbb{O}_T = \sigma_G (\mathbb{K}_O \mathbb{X}_T + \mathbb{V}_O \approx_{T-1} + \mathbb{S}_O)$
    $\mathbb{C}_T = \sigma_H (\mathbb{K}_C \mathbb{X}_T + \mathbb{V}_C \approx_{T-1} + \mathbb{S}_C)$
  Build the model using BiGRU and Softmax classifier
    $\overrightarrow{R_T} = \sigma \left(\overrightarrow{K_{XR}} X_T + \overrightarrow{K_{hR}} h_{T-1} + \overrightarrow{\mathbb{S}R}\right)$
    $\overrightarrow{Z_T} = \sigma \left(\overrightarrow{K_{XZ}} X_T + \overrightarrow{K_{hZ}} h_T - 1 + \overrightarrow{\mathbb{S}Z}\right)$
    $\overrightarrow{G_T} = \tanh \left(\overrightarrow{K_{XG}} X_T + \overrightarrow{K_{hG}} \left(\overrightarrow{R_T} \odot h_{T-1}\right) + \overrightarrow{\mathbb{S}G}\right)$
    $\overrightarrow{h_T} = \left(1 - \overrightarrow{Z_T}\right) \odot h_{T-1} + \overrightarrow{Z_T} \odot \overrightarrow{G_T}.$
    Add softmax layer
    $p (\hat{y}_c = y_c \mid x) = \varrho(x)_{y_c} = \frac{e^{x_c}}{\sum_j^C e^{x_j}}$
    Calculate categorical cross-entropy loss
    $L (\hat{y}_c, y_c) = -\sum_{i=1}^{n} \sum_{c=1}^{C} y_c^{x_i} \cdot \log (p (\hat{y}_{ic} = y_{ic} \mid x_i))$
  Perform Testing using Testset.
  Evaluate performance using various metrics
end procedure

LSTMVAE is compelled to learn the training time series data's most prominent characteristics. On the other hand, BiGRU gathers local dependencies in a two-way time flow, or forward and backward, to learn latent representations. Finally, the softmax layer is used to perform a multi-class attack detection process.

Each of the components mentioned, including the LSTMVAE, the BiGRU, and the softmax classifier, will be explained in detail in the following subsections.

2.3.1.1 LSTMVAE

Let the AE model takes Dataset $'\mathbb{X}'$ as an input and performs two major tasks, encoding ϕ and decoding ψ (Al-Hmouz, Pedrycz, Balamash & Morfeq (2022)), as described in Algorithm 1. These transition functions are given by the below equations.

$$\begin{aligned} \phi : \mathbb{X} \rightarrow \mathbb{K} := \approx = \sigma(\mathbb{K}\mathbb{X} + \mathbb{S}) \\ \psi : \mathbb{K} \rightarrow \hat{\mathbb{X}} = \sigma' \left(\mathbb{K}' \approx + \mathbb{S}'\right), \end{aligned} \tag{2.1}$$

where σ, $\mathbb{K}$, and $\mathbb{S}$ denote the activation function, weight matrix, and bias for encoder ϕ, respectively, and σ', $\mathbb{K}'$, and $\mathbb{S}'$ denote the activation function, weight matrix, and bias for decoder ψ, respectively. The AE uses backpropagation through time approach to solve the minimization problem formulated as:

$$\phi, \psi = \underset{\phi,\psi}{\arg\min} \|\mathbb{X} - (\psi \circ \phi)\mathbb{X}\|. \tag{2.2}$$

The AE calculates the Euclidean distance-based reconstruction error to generate representative features for reconstruction. This is done by the below equation.

$$\mathbb{L}(\mathbb{X}, \hat{\mathbb{X}}) = \|\mathbb{X} - \sigma' \left(\mathbb{K}'(\sigma(\mathbb{K}\mathbb{X} + \mathbb{S})) + \mathbb{S}'\right)\|^2. \tag{2.3}$$

Equation 2.1 is regarded as stateless since it can only capture duration-based aspects of the normalized likelihood sequence. It is also vulnerable to the vanishing gradient problem when the weight $\mathbb{K}$ is back-propagated over time (Chen, Du & Liao (2022)). As network traffic might have an unknown length in a real-time IoV network without experiencing any temporary changes, there may be random gaps in training activities. This can be solved by performing the encoding and decoding with LSTM units, i.e., replacing the latent representation $\approx$ in Equation 2.1 with Equation 2.4.

$$\approx_t = \mathbb{O}_T \circ \sigma_h \left(\mathbb{C}_T\right), \tag{2.4}$$

where $\mathbb{O}_T$ and $\mathbb{C}_T$ denote the output gate and cell state of h LSTM units, respectively. The relational composition between the previous and antecedent functions is indicated by the symbol $\circ$.

$$\mathbb{C}_T = (\mathbb{F}_T \circ \mathbb{C}_{T-1}) + \left(\mathbb{I}_T \circ \tilde{\mathbb{C}}_{T-1}\right), \tag{2.5}$$

where a forget gate activation $\mathbb{F}_T$, cell input activation $\tilde{\mathbb{C}}_T$, and update activation $\mathbb{I}_T$, control the state of the cell using the above equation.

Moreover, Equations 2.4 and 2.5 calculate the LSTM unit using the below transitions functions (Shu, Zhang, Sun & Tang (2021)):

$$\mathbb{F}_T = \sigma_G \left(\mathbb{K}_F \mathbb{X}_T + \mathbb{V}_F \approx_{T-1} + \mathbb{S}_F\right) \tag{2.6}$$

$$\mathbb{I}_T = \sigma_G \left(\mathbb{K}_I \mathbb{X}_T + \mathbb{V}_I \approx_{T-1} + \mathbb{S}_I\right) \tag{2.7}$$

$$\mathbb{O}_T = \sigma_G \left(\mathbb{K}_O \mathbb{X}_T + \mathbb{V}_O \approx_{T-1} + \mathbb{S}_O\right) \tag{2.8}$$

$$\mathbb{C}_T = \sigma_H \left(\mathbb{K}_C \mathbb{X}_T + \mathbb{V}_C \approx_{T-1} + \mathbb{S}_C\right), \tag{2.9}$$

where $\mathbb{K}$, $\mathbb{V}$, and $\mathbb{S}$ denote the weight matrix of the input $\mathbb{X}$, the weight matrix between the recurrent connections, and the bias parameters, respectively. These parameters are learned during the training process of LSTMVAE. The activation functions σ_G and σ_H are calculated using the below equations.

$$\sigma_G = S(X) = \frac{e^X}{e^X + 1} \tag{2.10}$$

$$\sigma_H = \tanh X = \frac{e^{2X} - 1}{e^{2X} + 1}. \tag{2.11}$$

Thus, by limiting the latent space to have a lower dimension compared to the input, the LSTMVAE is compelled to focus on learning the most critical aspects of the training set.

2.3.1.2 BiGRU

The latent features generated by the LSTMVAE are further passed to BiGRU, which gathers local dependencies in a two-way time flow, or forward and backward, to learn latent representations. The BiGRU has only two gate structures, reset gate and update gate. The transition functions of the BiGRU cell state are calculated using the below equations (Deng, Wang, Jia, Tong & Li (2019)). The reset gate primarily controls how the past information is merged with the current input information.

$$\overrightarrow{R_T} = \sigma \left(\overrightarrow{K_{XR}} X_T + \overrightarrow{K_{hR}} h_{T-1} + \overrightarrow{\mathbb{S}R}\right). \tag{2.12}$$

The update gate primarily controls how much prior information is kept in memory.

$$\overrightarrow{Z_T} = \sigma\left(\overrightarrow{K_{XZ}}X_T + \overrightarrow{K_{hZ}}h_T - 1 + \overrightarrow{\mathbb{S}Z}\right). \tag{2.13}$$

A candidate gate contains potential values that might be added to the cell state.

$$\overrightarrow{G_T} = \tanh\left(\overrightarrow{K_{XG}}X_T + \overrightarrow{K_{hG}}\left(\overrightarrow{R_T} \odot h_{T-1}\right) + \overrightarrow{\mathbb{S}G}\right) \tag{2.14}$$

$$\overrightarrow{h_T} = \left(1 - \overrightarrow{Z_T}\right) \odot h_{T-1} + \overrightarrow{Z_T} \odot \overrightarrow{G_T}, \tag{2.15}$$

where $\overrightarrow{R_T}$, $\overrightarrow{Z_T}$, and $\overrightarrow{G_T}$ denote reset gate, update gate, and candidate cell state, respectively. $\overrightarrow{K_{XR}}$, $\overrightarrow{K_{XZ}}$, $\overrightarrow{K_{XG}}$, $\overrightarrow{K_{XG}}$, and $\overrightarrow{K_{hG}}$ are the weight matrices. $\overrightarrow{\mathbb{S}R}$, $\overrightarrow{\mathbb{S}Z}$, and $\overrightarrow{R_T}$ are the bias vectors. The final state $\overrightarrow{h_T}$ is calculated by performing elementwise product $\odot$ between $(1 - \overrightarrow{Z_T})$ and (h_{T-1}), and performing elementwise product $\odot$ between $(\overrightarrow{Z_T})$ and $(\overrightarrow{G_T})$. Finally, at time T, BiGRU calculates its final output by using the previous frame at time $(T - 1)$ and the upcoming frame at time $(T + 1)$ using the below equation:

$$h_T = \overrightarrow{h_{T-1}} \oplus \overleftarrow{h_{T+1}}, \tag{2.16}$$

where $\oplus$ denotes the elementwise summation for forward and backward vectors.

2.3.1.3 Softmax classifier

Finally, we have used the softmax classifier as the last layer to perform multi-classification. Let us assume that the proposed LSTMVAE-BiGRU-based IDS has the last layer as a softmax function. Following that, this layer is given the input sequence $X = \left(X_1, X_2, \ldots X_T^N\right)$, and the output layer of the network produces a one-hot encoded C-dimensional vector y while carrying across T timesteps. Hence, the following formula is used to determine the probability that a single input, X, relates to a certain threat category, y (Al-Hawawreh *et al.* (2020)).

$$p\left(\hat{y}_c = y_c \mid X\right) = \varrho(X)_{y_c} = \frac{e^{X_c}}{\sum_j^C e^{X_j}} \quad (C = 1, 2, \ldots c) \tag{2.17}$$

$$L\left(\hat{y}_c, y_c\right) = -\sum_{i=1}^{n}\sum_{c=1}^{C} y_c^{P_i} \cdot \log\left(p\left(\hat{y}_{ic} = y_{ic} \mid P_i\right)\right). \tag{2.18}$$

The loss during the training procedure is calculated using the categorical cross-entropy loss function, as given above.

2.4 Conclusion

In this chapter, we have provided a comprehensive overview of our system model, which serves as a solution to detect and classify various cyber-attacks in IoV networks. Furthermore, we have presented a detailed description of the hybrid DL model used in our proposed IDS. This model combines three key components: the LSTMVAE, the BiGRU, and the softmax classifier. We have explained the purpose and functionality of each component and how they contribute to the overall performance of the IDS. Lastly, we have provided mathematical representations for each of the three components in our IDS architecture.

CHAPTER 3

RESULTS

3.1 Introduction

In this chapter, we will delve into the technical aspects of our methodology. We will start by providing a comprehensive explanation of our ToN-IoT dataset, which includes a detailed description of the preprocessing steps applied to the dataset.

Next, we will introduce and discuss the different evaluation metrics that we have employed to assess the performance of our proposed IDS model. Following that, we will present the experimental setup that we have used to train our hybrid DL model. This will include information on the parameters and hyperparameters chosen for the model training process. Finally, we will analyze the performance of our solution using various metrics and compare it against commonly used baseline techniques and some recent state-of-the-art methods.

3.2 Dataset Description

The ToN-IoT dataset (Moustafa (2021)) is introduced by the UNSW Canberra IoT Labs and the School of Engineering and Information Technology at UNSW Canberra to evaluate the efficiency of several AI-enabled intrusion detection applications. To enhance the security of the IoV network, we have developed an IDS model that utilizes the ToN-IoT dataset. The ToN-IoT dataset consists of a diverse range of cyber-security attacks, encompassing various types such as backdoor attacks, DDoS, DoS, injection attacks, MITM attacks, password attacks, ransomware attacks, scanning attacks, and XSS attacks.

Along with the attack labels, the dataset comprises 44 distinct features that capture relevant information for detecting and classifying these attacks. Additionally, the dataset includes a normal class, which represents benign or non-malicious network traffic, providing a baseline for comparison and enabling the identification of anomalous patterns associated with attacks.

The process begins with preprocessing the dataset and is followed by a series of steps as follows: Imputation of missing values with the mean of that particular row, followed by converting categorical features into numeric data points using the label-encoding technique. Then, the dataset values were normalized between 0 and 1 using the min-max scaler technique. The ToN-IoT dataset is then divided into a 70-30 ratio of training and testing set. Finally, the proposed approach is compared with some baseline techniques, i.e., Random Forest (Baseline-1), Decision Trees (Baseline-2), and Naive Bayes (Baseline-3), that are commonly used in the literature.

3.3 Performance Metrics

The performance of the proposed IDS is evaluated using different evaluation metrics. For instance, we have used False Negative (α), True Negative (β), False Positive (γ), and True Positive (δ) parameters to compute different metrics such as Precision Rate (PR), Detection Rate (DR), Accuracy (AC), False Alarm Rate (FAR), and F1 Score, which are defined as follows:

- **Precision:** The metric measures the ratio of the samples correctly identified as positive by the total positive identified samples, correct or incorrect, as follows:

$$Precision(PR) = \frac{\delta}{\delta + \gamma} \tag{3.1}$$

- **Detection Rate:** It is defined as the ratio of the samples correctly predicted as positive by the total positive identified samples, either correctly identified as positive or incorrectly identified as negative, as follows:

$$DetectionRate = \frac{\delta}{\alpha + \delta} \tag{3.2}$$

- **Accuracy:** The metric is defined as the sum of the number of correctly identified predictions as positive or negative to all type of predictions including correct and incorrect, as follows:

$$Accuracy = \frac{\delta + \beta}{\gamma + \beta + \delta + \alpha} \tag{3.3}$$

- **False Alarm Rate:** The metric measures the ratio of the samples incorrectly identified as positive by the total number of negatives, either correctly identified as negative or incorrectly identified as positive, as follows:

$$FalseAlarmRate = \frac{\gamma}{\gamma + \beta} \tag{3.4}$$

- **F1 Score:** This metric indicates the harmonic mean of PR and Recall (RC), which can be formulated as follows:

$$F1Score = 2 * \frac{PR * RC}{PR + RC} \tag{3.5}$$

The highest and the lowest possible value of an F1 Score is 1 and 0, respectively.

3.4 Simulation Environment

The proposed approach is implemented using the Tensorflow library and Python 3.5. The experimental setup to perform the simulations are mentioned in Table 3.1.

Table 3.1 Hyper-parameters used for designing LSTMVAE-BiGRU-based IDS

Settings	**Hyperparameters**
Input Layer	44 attributes from the ToN-IoT dataset
Encoder	Hidden Layers (HL)=4; Hidden Nodes (HN)= (44,35,25,10), tanh function
Decoder	HL=4; HN= (10,25,35,44), 3 Layers use tanh, and the last layer uses sigmoid
LSTMVAE Model	loss='binary crossentropy', optimizer='adam', batch size=50, epochs=10
Hidden layers	HL=5; HN= (30,128,64,15,5)
Output layer	a softmax function 1 normal and 9 attacks
BiGRU Model	optimizer='adam', loss='categorical crossentropy', batch size=50, epochs=10

3.5 Results

This section demonstrates the implementation details of the proposed model. We analyze the performance on the basis of accuracy vs. loss and confusion matrix. Moreover, we perform per-class prediction analysis to perform a multi-class attack detection process. Afterwards, we perform the comparison analysis against commonly used baseline techniques in terms of accuracy, precision, detection rate, and F1 score. Finally, the efficacy of the proposed IDS model is assessed by comparing it with recent state-of-the-art methods.

3.5.1 Performance Analysis

Within this subsection, we conduct a thorough analysis of the proposed IDS on various metrics. A detailed explanation is described below.

3.5.1.1 Accuracy vs. loss

The proposed LSTMVAE technique has efficiently learned the dataset. The hyperparameters used to design the LSTMVAE technique are mentioned in Table 3.1. In Fig. 3.1, we show the outcome in terms of accuracy vs. loss, and the IDS obtained a 0.0872% *validation loss* and 0.9649% *validation accuracy*. The LSTMVAE technique's objective is not to discover these threat observations but to extract important low-dimensional attributes. Thus, the extracted features are used by the BiGRU with a softmax classifier to detect different attacks present in the dataset.

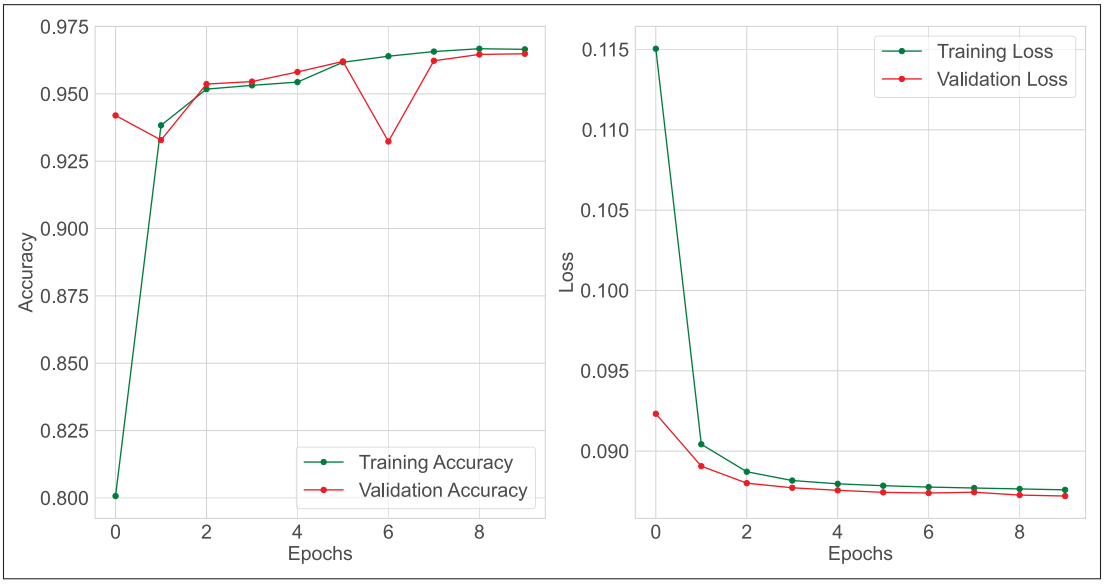

Figure 3.1 Accuracy vs. loss for LSTMVAE

3.5.1.2 Confusion matrix

The Confusion Matrix (CM) is a useful tool for assessing the performance of an IDS in binary classifications. All four possible outcomes γ, α, δ, and β are represented in the matrix. In this context, positive indicates that the IDS identified the traffic record as an attack, while negative indicates that the IDS classified it as normal data. The diagonal elements in Fig. 3.2 show the correct classification of various attack vectors and a normal class. It is concluded from the confusion matrix that the false-positive rate for each class in the ToN-IoT dataset is very low, and thus the proposed IDS has a high detection rate.

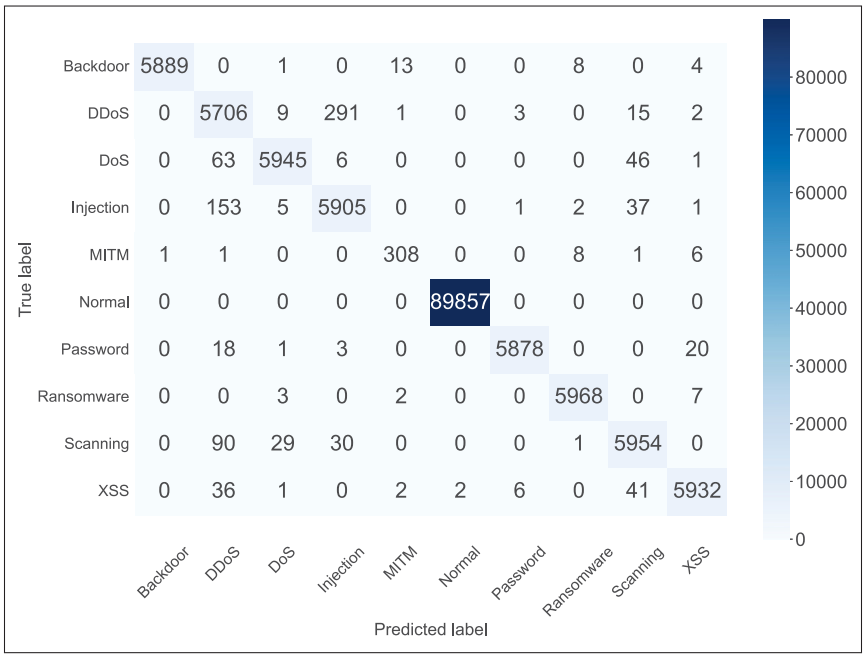

Figure 3.2 Confusion matrix for the proposed model

3.5.1.3 Per-class prediction

The per-class prediction analysis in terms of PR, DR, F1 score, and FAR is illustrated in Table 3.2. This analysis is useful when IDS performs a multi-class attack detection process. It can be concluded from Table 3.2 that the proposed IDS has achieved PR, DR, and F1 score values

between 94%-100%. On the other hand, the false alarm for each class in the ToN-IoT dataset is very low, i.e., 0%, which indicates the success rate of the proposed IDS.

Table 3.2 Per-class prediction results (%) for LSTMVAE-BiGRU-based IDS

Parameters	Backdoor	DDoS	DoS	Injection	MITM	Normal	Password	Ransomware	Scanning	XSS
PR	99.99	94.04	99.18	94.70	94.47	99.99	99.83	99.68	97.70	99.31
DR	99.56	94.67	98.08	96.73	94.76	100.00	99.29	99.79	97.54	98.53
F1	99.77	94.36	98.63	95.71	94.62	99.99	99.55	99.74	97.62	98.92
FAR	0.000007	0.002728	0.000370	0.002496	0.000130	0.000041	0.000075	0.000143	0.001058	0.000309

3.5.1.4 ROC Curve

The Receiver Operating Characteristic (ROC) is a useful method for interpreting the results of multi-class vectors in the dataset. As a visual assessment tool, ROC graphs are particularly useful for assessing the efficacy of classifiers. The δ and γ axes are typically extended within the bounds of the two-dimensional ROC space. Based on Fig. 3.3, the following details are deduced: the reported micro-average Area under the ROC Curve (AUC) for the LSTMVAE-BiGRU-based IDS is significantly high, i.e., 0.99996, and the macro-average AUC value is 0.99977. On the other hand, it can be seen that the AUC values for the different attacks and a normal class are between 0.99887 to 1.00. This indicates that the proposed IDS can easily distinguish between different attacks and a normal class in the ToN-IoT dataset.

3.5.2 Comparison with baselines techniques

In this subsection, we compare the performance of the proposed LSTMVAE-BiGRU model against commonly used baseline techniques. First, we compare the performance in terms of per-class DR (multi-class attack detection) scenario. Table 3.3 shows the comparison for various attacks and a normal class present in the dataset. It is worth noting that the values for DR metrics are between 94%-100% for LSTMVAE-BiGRU-based IDS. On the other hand, Baseline-2 has achieved higher values for most attacks, but for injection and MITM attacks, the DR is 0%. This table indicates that the proposed IDS has outperformed its competitive models.

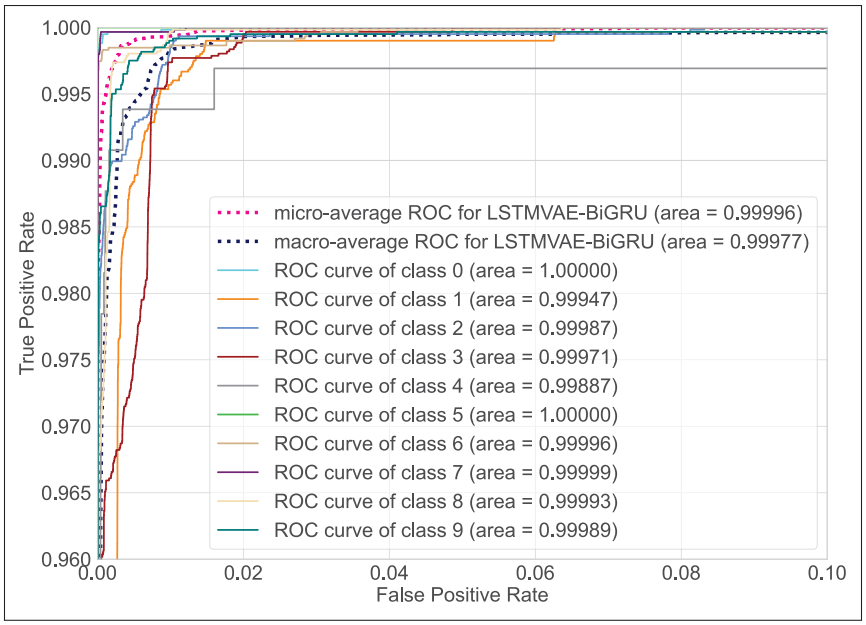

Figure 3.3 ROC for the proposed LSTMVAE-BiGRU-based IDS

Table 3.3 Comparison of DR with baseline techniques

Techniques	Backdoor	DDoS	DoS	Injection	MITM	Normal	Password	Ransomware	Scanning	XSS
Baseline-1	99.98	90.40	91.97	93.53	0.00	100.00	97.81	99.40	95.74	85.47
Baseline-2	100.00	100.00	100.00	0.00	0.00	100.00	100.00	100.00	100.00	100.00
Baseline-3	99.22	26.80	91.70	92.96	95.11	100.00	75.32	79.98	96.91	19.02
Proposed IDS	99.56	94.67	98.08	96.73	94.76	100.00	99.29	99.79	97.54	98.53

Finally, we have compared the performance of the proposed IDS with baseline techniques in terms of accuracy, precision, detection rate, and F1 score. Fig. 3.4 shows this comparison. It can be concluded that the proposed IDS has achieved higher values for these parameters. We have also compared the performance of the proposed IDS with some recent state-of-the-art methods Alsaedi, Moustafa, Tari, Mahmood & Anwar (2020); Abdel-Basset *et al.* (2022); Oseni *et al.* (2023); Booij, Chiscop, Meeuwissen, Moustafa & Hartog (2022) using accuracy metrics. Table 3.4 highlights this comparison, and we may conclude that LSTMVAE-BiGRU-based IDS performs better in recognizing the cyber-attacks for the IoV networks.

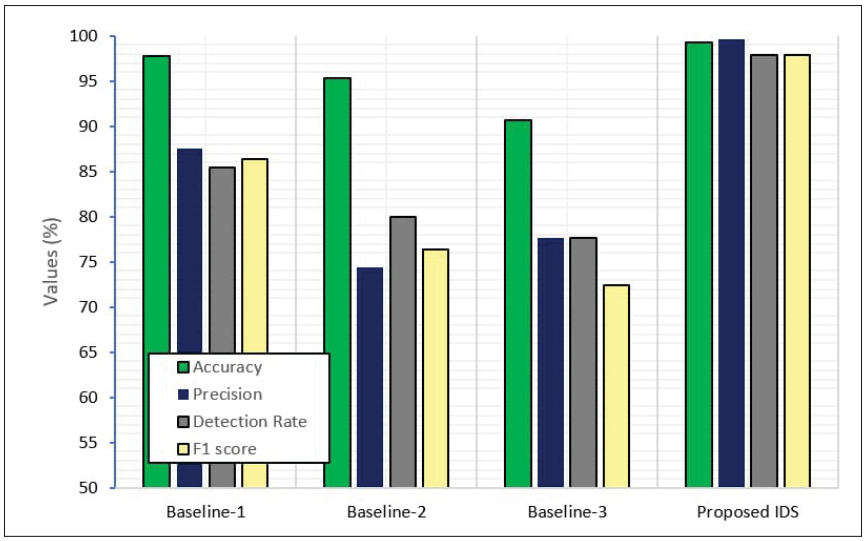

Figure 3.4 Performance comparison with baseline techniques

Table 3.4 Performance comparison with other models

Works	Year	Method	Data Source	Accuracy
Alsaedi *et al.* (2020)	2021	Classification And Regression Tree	ToN-IoT	88.00%
Abdel-Basset *et al.* (2022)	2022	Federated IDS	ToN-IoT	94.85%
Booij *et al.* (2022)	2022	Random Forest	ToN-IoT	98.07%
Oseni *et al.* (2023)	2023	Convolutional Neural Networks	ToN-IoT	90.55%
Proposed IDS	2023	LSTMVAE-BiGRU	ToN-IoT	99.30%

3.6 Discussion

The evaluation results from the previous section provide insights into the proposed solution presented in this project:

- **Architecture:** The proposed architecture combines statistical feature extraction with the robust learning mechanism of the LSTMVAE paradigm, along with BiGRU and a softmax layer. This integration aims to create an effective hybrid DL model.

- **Feature Extraction:** The LSTMVAE plays a crucial role in feature extraction by extracting low-dimensional attributes. These extracted features are then fed to the BiGRU with a softmax classifier, enabling the model to learn latent representations and perform attack classification. This feature extraction process contributes to improving the overall performance of the model.
- **Accuracy Comparison:** The proposed LSTMVAE-BiGRU algorithm outperforms baseline techniques such as Random Forest, Decision Trees, and Naive Bayes in terms of accuracy. Additionally, when compared with recent state-of-the-art methods like Classification And Regression Tree, Federated IDS, random forest, and CNNs, the proposed technique achieves higher accuracy.
- **Attack Detection:** The proposed IDS demonstrates strong performance in both binary classification and multi-class attack detection. Unlike systems that only detect specific types of attacks, the proposed model can detect different attack types effectively. Moreover, it achieves a low FAR while maintaining higher PR, DR, and F1 score, indicating its superior performance.

3.7 Conclusion

Based on the simulation results, our proposed LSTMVAE-BiGRU algorithm has proven to be highly effective in identifying and classifying potential cyber-attacks within the IoV network. The LSTMVAE-BiGRU algorithm outperforms commonly used baseline techniques, showing superior AC, PR, DR, while maintaining a low FAR. Furthermore, the proposed algorithm also outperforms state-of-the-art approaches in terms of accuracy.

It can be concluded that the proposed hybrid DL model is a robust and reliable solution for detecting and classifying cyber-attacks within the IoV network, offering superior performance compared to baseline techniques and state-of-the-art approaches.

CONCLUSION AND RECOMMENDATIONS

The emergence of smart cities and advancements in telecommunication technologies have made the IoV more vulnerable to cyber threats. The lack of sufficient security implementations in IoVs exposes them to various network attacks. In order to protect IoVs, ensure the safety of individuals, and prevent physical harm, IDSs offer effective solutions.

The goal of this dissertation was to propose a solution for identifying and classifying potential cyber-attacks in the IoV network. The proposed approach involved developing a novel IDS based on a generative hybrid DL model. The proposed model focused on learning time series and multivariate data from the IoV network, distinguishing between trustworthy and uncertain instances by leveraging the most significant attributes.

The proposed IDS, based on LSTMVAE-BiGRU architecture, consists of three main components: LSTMVAE, BiGRU, and the softmax classifier. Within this architecture, the LSTM network is responsible for encoding and decoding using the LSTMVAE paradigm, which combines LSTM with AE capabilities. By constraining the latent space to be smaller than the input dimension, the LSTMVAE is compelled to capture the most significant characteristics of the training time series data. Additionally, BiGRU captures local dependencies in both forward and backward time flow, enabling the learning of latent representations. Finally, the softmax layer performs the multi-class attack detection process. The experimental results substantiate that the proposed IDS outperformed widely employed baseline techniques and state-of-the-art approaches. It has demonstrated superior performance, achieving higher values in terms of accuracy, precision, detection rate, and F1 score.

Future research directions include testing the proposed IDS on diverse datasets to evaluate its effectiveness and performance across various scenarios and data characteristics. Developing an IDS that can operate effectively in large-scale IoV environments is a significant challenge and an important area for further investigation. Additionally, addressing the detection of complex

attack scenarios, such as zero-day attacks, would contribute to strengthening the security of IoV systems and mitigating emerging threats.

APPENDIX I

LSTM-BASED HYBRID INTRUSION DETECTION SYSTEM ARTICLE

This article, entitled "LSTM-Based Hybrid Intrusion Detection System for Internet of Vehicles" has been accepted for publication in the proceedings of the IEEE Global Communications Conference (GLOBECOM 2023, Kuala Lumpur, Malaysia).

LSTM-Based Hybrid Intrusion Detection System for Internet of Vehicles

Kanika Aggarwal* and Georges Kaddoum†
*†Electrical Engineering Department, École de Technologie Supérieure (ÉTS), Montreal, QC H3C 1K3, Canada.
†Cyber Security Systems and Applied AI Research Center, Lebanese American University, Lebanon.
Emails: [*kanika.aggarwal.1@ens.etsmtl.ca, †georges.kaddoum@etsmtl.ca]

***Abstract*—The recent growth of the Internet of Things (IoT) has revolutionized vehicular networks into the Internet of Vehicles (IoV). Within the IoV infrastructure, modern vehicles are vulnerable to different and new types of cyber-attacks. Consequently, Intrusion Detection Systems (IDS) are extremely helpful in coping with such attacks, and there is a pressing need for the development of an advanced IDS that can efficiently detect attacks with a high detection rate and accuracy, and a low false alarm rate. Toward this end, we present a deep learning-based IDS for the identification and classification of potential cyber-attacks in the IoV network. Specifically, a generative hybrid deep learning model that combines Long Short-Term Memory Variational AutoEncoder with Bidirectional Gated Recurrent Units (LSTMVAE-BiGRU) is proposed. Due to the integration of the LSTM network and the VAE in this architecture, the LSTM performs the encoding and decoding in the LSTMVAE paradigm. The LSTMVAE is forced to learn the most salient features of the training time series data by restricting the latent space to have a dimension that is less than the input. On the other hand, to learn latent representations, BiGRU collects local dependencies in a two-way time flow, or forward and backward. Lastly, the multi-class attack detection method is carried out using the softmax layer. Experimental findings demonstrate that the proposed IDS outperforms both baseline techniques that are widely used and cutting-edge approaches.**

***Index Terms*—Intrusion Detection System, Internet of Vehicles (IoV), security, Long Short-Term Memory, deep learning**

I. Introduction

With the rapid advances in the IoT, IoV has emerged as a disruptive technology. Within the IoV infrastructure, modern vehicles are equipped with advanced onboard sensors and networking capabilities connected to the Internet. The sensors embedded in modern vehicles collect road data and allow them to communicate and share with other vehicles, smart devices, and surrounding objects through vehicle-to-everything (V2X) wireless communications. V2X can further take the form of vehicle-to-infrastructure (V2I), vehicle-to-human (V2H), vehicle-to-network (V2N), and vehicle-to-vehicle (V2V) technologies [1]. These technologies have paved the way for enhanced driving experience and intelligent transportation.

Along with the services offered, IoV plays a crucial role in supporting a wide range of applications, including autonomous vehicles (AVs), traffic and parking management, tracking systems, infotainment, etc. According to Statista [2], it is expected to have around 146 million AVs in the United States by 2030. However, the massive growth in AVs results in an increased number of V2I and V2V communication links, which poses a major challenge to securing the IoV network. Furthermore, IoV is an open and dynamic heterogeneous network that interconnects smart vehicles, nearby surroundings, and public internet connections. This growing connectivity makes them vulnerable to multiple types of cyber-attacks, thus; threatening the trust, privacy, and security of modern vehicles and causing serious risks to human lives [3].

The most common cyber-attacks on IoV comprise ransomware, backdoor, password attack, denial of service (DoS), man-in-the-middle (MITM), scanning, cross-site scripting (XSS), data injection, and distributed denial of service (DDoS) [3]. In this direction, IDS is considered a powerful mean to detect possible cyber-security attacks that compromise the integrity, confidentiality, availability, authenticity, reliability, and privacy of vehicular networks. To address the aforementioned security issues, several existing works presented in the literature use artificial intelligence-based IDS techniques. For example, Anzar *et al.* [4] designed a supervised learning-based IDS to efficiently identify the intruders in the IoV network based on V2V communications. However, they trained their model based on the KDD Cup 1999 dataset, which is unsuitable for vehicular networks. Sherazi *et al.* [5] designed a fuzzy logic-based framework and implemented it on a 6BR device to identify and prevent DDoS attacks in IoV communications.

Similarly, focusing on malicious security failures, the authors in [6] proposed a distributed Machine Learning (ML) scheme for software-defined IoV networks. Yang *et al.* in [7] proposed a novel ensemble-based IDS model for IoV networks using ML approaches. However, most of these security solutions suffer from low detection accuracy, high complexity, and lack of generalization capabilities, making them unsuitable for dynamic attacks. Therefore, considering the high connectivity, the complexity of IoV network topology, constraints of storage and computing resources, and multiple cyber-attacks being launched, the traditional IDS mechanisms are not applicable in real-time scenarios.

Owing to the explosion of vehicular traffic data, the Deep Learning (DL) models [3] have gained significant attention from academia and the telecom industry due to their potential to effectively deal with heterogeneous, unstructured, and large volumes of data. For instance, the work in [8] deployed the detection engines on multi-access edge computing

978-1-6654-3540-6/22

(MEC) servers in IoV networks. A DL-based IDS framework is proposed, which comprises DL engines to identify and classify malicious traffic. However, the proposed approach has considered fewer cyber-security attacks for simulation. Oseni *et al.* [9] employed a Shapley Additive exPlanations (SHAP) scheme to study the output of a DL-based IDS in IoV systems. However, the considered approach is computationally expensive and vulnerable to adversarial attacks.

A convolutional neural network (CNN)-based IDS is proposed by Nie *et al.* [10] to detect different intrusion attacks that aim at Road Side Units (RSUs). However, a limited set of attacks were considered in their work. Another approach in [11], where Ahmed *et al.* tried to solve the anomaly detection problem in the controller area network (CAN) bus by leveraging a CNN, specifically the VGG-16 architecture. The model is trained using the CAN-intrusion dataset, which includes fuzzy attacks, DoS attacks, and normal attacks. Almutlaq *et al.* [12] used the rule extraction method from deep neural networks and implemented a two-stage IDS for intelligent transportation systems.

Although different DL approaches have been proposed in the literature, most yield low detection accuracy and high false alarm rates. These techniques also face key challenges. Due to the fast movement of modern vehicles, the topology of IoV changes frequently and thus they access the network randomly. On the other hand, for intrusion detection, DL-based systems frequently, require a large amount of network data to discover abnormalities. However, getting such an enormous amount of data is a challenging issue. In addition, extracting the features of all possible patterns available in the IoV network is a challenging task since these features constitute the model's overall performance.

To address the challenges, recent studies have proposed Long Short-Term Memory (LSTM) architecture as one of the promising solutions for IDS. LSTMs are a special kind of recurrent neural networks (RNNs) that can handle long-term dependencies [1]. The key aspect of LSTM-based RNNs is to remember the information for long periods, which makes them suitable for performing experiments on spatial data that changes over time. For instance, Ashraf *et al.* [13] presented a LSTM autoencoder-based intrusion detection framework that can identify abnormal behaviour in both external networks and CANs. Another DL-based threat intelligence scheme has been presented in [14] to detect cyber-attacks from space-air-ground-sea (SAGS) networks. A deep stacked autoencoder (DSAE) was used to extract the hidden patterns of IoT-SAGS network traffic, and then, a Gated Recurrent Neural Network was applied to detect the cyber-attack types. However, these approaches are designed for modeling the univariate sequence representations. Thus, we propose a novel IDS for the identification and classification of potential cyber-attacks in the IoV network. In our work, a hybrid DL model that combines LSTM Variational AutoEncoder with Bidirectional Gated Recurrent Units to learn time series and multivariate data from IoV networks is presented.

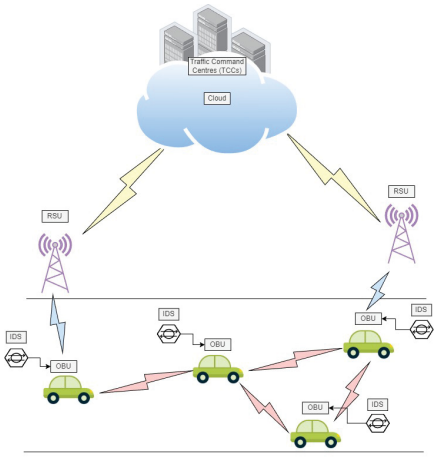

Fig. 1: Network model of proposed IDS.

II. Contribution

The main contributions of this article are as follows:

- A novel IDS based on a generative hybrid deep learning model is proposed.
- Specifically, a statistical feature extraction technique based on LSTMVAE is designed. The proposed LSTM-VAE can learn time series and multivariate data from the IoV network efficiently. Second, the extracted features are used by the BiGRU and softmax classifier to differentiate between reliable and doubtful occurrences in the IoV network.
- The performance of the proposed IDS is evaluated using various evaluation metrics and further compared with some commonly used baseline techniques and state-of-the-art methods using the ToN-IoT dataset [15].

III. Proposed Model

A. Working Architecture of the proposed IDS

Fig. 1 shows the network model for the proposed IDS. The network model contains three entities: Traffic Command Centres (TCCs) located at cloud servers, Road Side Units (RSUs), and On-Board Units (OBUs). Typically, in the IoV network, vehicles are equipped with OBU units that record traffic and driving data, and share this information with RSUs or other nearby vehicles in the network. The RSUs are positioned at different geographical locations and share the vehicles real-time information with TCCs. An attacker can easily target the OBUs located on the top of vehicles and can further attempt to exploit the telematics services of the vehicles. Since most vehicles have enough memory and computer power, we recommend installing the proposed IDS as a firmware or software within the OBUs. Specifically, the proposed IDS is a generative hybrid DL model that combines LSTMVAE-BiGRU to learn time series and multivariate data from the IoV network, and is capable of differentiating between reliable and doubtful instances based on the most representative attributes.The LSTMVAE approach is designed to ensure feature extraction, and BiGRU with softmax classifier is combined for

attack classification. The proposed architecture is explained below:

B. LSTMVAE-BiGRU-based IDS

The proposed LSTMVAE-BiGRU-based IDS has three main parts: LSTMVAE, BiGRU, and the softmax classifier. In this architecture, the LSTM performs the encoding and decoding in the LSTMVAE paradigm because it combines the LSTM network with the AutoEncoder (AE). Furthermore, by limiting the latent space to have a dimension that is less than the input, the LSTMVAE is compelled to learn the training time series data's most prominent characteristics. On the other hand, BiGRU gathers local dependencies in a two-way time flow, or forward and backward, to learn latent representations. Finally, the softmax layer is used to perform a multi-class attack detection process. Each of them is explained below:

1) LSTMVAE: Let the AE model takes Dataset $'\mathbb{X}'$ as an input and performs two major tasks, encoding ϕ and decoding ψ [16], as described in Algorithm 1. These transition functions are given by the below equations.

$$\begin{aligned} \phi : \mathbb{X} \rightarrow \mathbb{K} := \mathbb{h} = \sigma(\mathbb{K}\mathbb{X} + \mathbb{S}) \\ \psi : \mathbb{K} \rightarrow \hat{\mathbb{X}} = \sigma'\left(\mathbb{K}'\mathbb{h} + \mathbb{S}'\right), \end{aligned} \tag{1}$$

where σ, $\mathbb{K}$, and $\mathbb{S}$ denote the activation function, weight matrix, and bias for encoder ϕ, respectively, and σ', $\mathbb{K}'$, and $\mathbb{S}'$ denote the activation function, weight matrix, and bias for decoder ψ, respectively. The AE uses backpropagation through time approach to solve the minimization problem formulated as:

$$\phi, \psi = \underset{\phi,\psi}{arg\,min} \|\mathbb{X} - (\psi \circ \phi)\mathbb{X}\|. \tag{2}$$

The AE calculates the Euclidean distance-based reconstruction error to generate representative features for reconstruction. This is done by the below equation.

$$\mathbb{L}(\mathbb{X}, \hat{\mathbb{X}}) = \left\|\mathbb{X} - \sigma'\left(\mathbb{K}'(\sigma(\mathbb{K}\mathbb{X} + \mathbb{S})) + \mathbb{S}'\right)\right\|^2. \tag{3}$$

Equation 1 is regarded as stateless since it can only capture duration-based aspects of the normalized likelihood sequence. It is also vulnerable to the vanishing gradient problem when the weight $\mathbb{K}$ is back-propagated over time [17]. As network traffic might have an unknown length in a real-time IoV network without experiencing any temporary changes, there may be random gaps in training activities. This can be solved by performing the encoding and decoding with LSTM units, i.e., replacing the latent representation $\mathbb{h}$ in Equation 1 with Equation 4.

$$\mathbb{h}_t = \mathbb{O}_T \circ \sigma_h\left(\mathbb{C}_T\right), \tag{4}$$

where $\mathbb{O}_T$ and $\mathbb{C}_T$ denote the output gate and cell state of h LSTM units, respectively. The relational composition between the previous and antecedent functions is indicated by the symbol $\circ$.

$$\mathbb{C}_T = \left(\mathbb{F}_T \circ \mathbb{C}_{T-1}\right) + \left(\mathbb{I}_T \circ \tilde{\mathbb{C}}_{T-1}\right), \tag{5}$$

where a forget gate activation $\mathbb{F}_T$, cell input activation $\tilde{\mathbb{C}}_T$, and update activation $\mathbb{I}_T$, control the state of the cell using the above equation.

Algorithm 1 Proposed LSTMVAE-BiGRU

procedure INPUT:(Read the Dataset)
  OUTPUT: Attack classification
  Pre-process Dataset
    Imputation of missing values
    Convert categorical features into numeric data
    Normalize data values between 0 and 1
  Divide data into Training, Validation, and Testing sets.
  Feature Extraction using the LSTMVAE technique
    Perform encoding and decoding
    $\phi : \mathbb{X} \rightarrow \mathbb{K} := \mathbb{h} = \sigma(\mathbb{K}\mathbb{X} + \mathbb{S})$
    $\psi : \mathbb{K} \rightarrow \hat{\mathbb{X}} = \sigma'\left(\mathbb{K}'\mathbb{h} + \mathbb{S}'\right)$
    Replace the latent features with
    $\mathbb{h}_t = \mathbb{O}_T \circ \sigma_h\left(\mathbb{C}_T\right)$
    Add LSTM layers and perform encoding
    $\mathbb{C}_T = \left(\mathbb{F}_T \circ \mathbb{C}_{T-1}\right) + \left(\mathbb{I}_T \circ \tilde{\mathbb{C}}_{T-1}\right)$
    $\mathbb{F}_T = \sigma_G\left(\mathbb{K}_F\mathbb{X}_T + \mathbb{V}_F\mathbb{h}_{T-1} + \mathbb{S}_F\right)$
    $\mathbb{I}_T = \sigma_G\left(\mathbb{K}_I\mathbb{X}_T + \mathbb{V}_I\mathbb{h}_{T-1} + \mathbb{S}_I\right)$
    $\mathbb{O}_T = \sigma_G\left(\mathbb{K}_O\mathbb{X}_T + \mathbb{V}_O\mathbb{h}_{T-1} + \mathbb{S}_O\right)$
    $\mathbb{C}_T = \sigma_H\left(\mathbb{K}_C\mathbb{X}_T + \mathbb{V}_C\mathbb{h}_{T-1} + \mathbb{S}_C\right)$
  Build the model using BiGRU and softmax classifier
    $\overrightarrow{R_T} = \sigma\left(\overrightarrow{K_{XR}}X_T + \overrightarrow{K_{hR}}h_{T-1} + \overrightarrow{\mathbb{S}R}\right)$
    $\overrightarrow{Z_T} = \sigma\left(\overrightarrow{K_{XZ}}X_T + \overrightarrow{K_{hZ}}h_T - 1 + \overrightarrow{\mathbb{S}Z}\right)$
    $\overrightarrow{G_T} = \tanh\left(\overrightarrow{K_{XG}}X_T + \overrightarrow{K_{hG}}\left(\overrightarrow{R_T} \odot h_{T-1}\right) + \overrightarrow{\mathbb{S}G}\right)$
    $\overrightarrow{h_T} = \left(1 - \overrightarrow{Z_T}\right) \odot h_{T-1} + \overrightarrow{Z_T} \odot \overrightarrow{G_T}.$
    Add softmax layer
    $p\left(\hat{y}_c = y_c \mid x\right) = \varrho(x)_{y_c} = \frac{e^{x_c}}{\sum_j^C e^{x_j}}$
    Calculate categorical cross-entropy loss
    $L\left(\hat{y}_c, y_c\right) = -\sum_{i=1}^{n}\sum_{c=1}^{C} y_c^{x_i} \cdot \log\left(p\left(\hat{y}_{ic} = y_{ic} \mid x_i\right)\right)$
  Perform Testing using Testset.
  Evaluate performance using various metrics
end procedure

Moreover, Equations 4 and 5 calculate the LSTM unit using the below transitions functions [18]:

$$\mathbb{F}_T = \sigma_G\left(\mathbb{K}_F\mathbb{X}_T + \mathbb{V}_F\mathbb{h}_{T-1} + \mathbb{S}_F\right) \tag{6}$$

$$\mathbb{I}_T = \sigma_G\left(\mathbb{K}_I\mathbb{X}_T + \mathbb{V}_I\mathbb{h}_{T-1} + \mathbb{S}_I\right) \tag{7}$$

$$\mathbb{O}_T = \sigma_G\left(\mathbb{K}_O\mathbb{X}_T + \mathbb{V}_O\mathbb{h}_{T-1} + \mathbb{S}_O\right) \tag{8}$$

$$\mathbb{C}_T = \sigma_H\left(\mathbb{K}_C\mathbb{X}_T + \mathbb{V}_C\mathbb{h}_{T-1} + \mathbb{S}_C\right), \tag{9}$$

where $\mathbb{K}$, $\mathbb{V}$, and $\mathbb{S}$ denote the weight matrix of the input $\mathbb{X}$, the weight matrix between the recurrent connections, and bias parameters, respectively. These parameters are learned during the training process of LSTMVAE. The activation functions σ_G and σ_H are calculated using the below equations.

$$\sigma_G = S(X) = \frac{e^X}{e^X + 1} \tag{10}$$

$$\sigma_H = \tanh X = \frac{e^{2X} - 1}{e^{2X} + 1}. \tag{11}$$

Thus, by limiting the latent space to have a lower dimension than the input, the LSTMVAE is compelled to learn the most important aspects of the training set.

2) BiGRU: The latent features generated by the LSTMVAE are further passed to BiGRU, which gathers local dependencies in a two-way time flow, or forward and backward, to learn latent representations. The BiGRU has only two gate structures, reset gate and update gate. The transition functions of the BiGRU cell state are calculated using the below equations [19].

The reset gate primarily controls how the past information is merged with the current input information.

$$\overrightarrow{R_T} = \sigma\left(\overrightarrow{K_{XR}}X_T + \overrightarrow{K_{hR}}h_{T-1} + \overrightarrow{\$R}\right). \quad (12)$$

The update gate primarily controls how much prior information is kept in memory.

$$\overrightarrow{Z_T} = \sigma\left(\overrightarrow{K_{XZ}}X_T + \overrightarrow{K_{hZ}}h_T - 1 + \overrightarrow{\$Z}\right). \quad (13)$$

A candidate gate contains potential values that might be added to the cell state.

$$\overrightarrow{G_T} = \tanh\left(\overrightarrow{K_{XG}}X_T + \overrightarrow{K_{hG}}\left(\overrightarrow{R_T} \odot h_{T-1}\right) + \overrightarrow{\$G}\right) \quad (14)$$

$$\overrightarrow{h_T} = \left(1 - \overrightarrow{Z_T}\right) \odot h_{T-1} + \overrightarrow{Z_T} \odot \overrightarrow{G_T}, \quad (15)$$

where $\overrightarrow{R_T}$, $\overrightarrow{Z_T}$, and $\overrightarrow{G_T}$ denote reset gate, update gate, and candidate cell state. $\overrightarrow{K_{XR}}$, $\overrightarrow{K_{XZ}}$, $\overrightarrow{K_{XG}}$, $\overrightarrow{K_{XG}}$, and $\overrightarrow{K_{hG}}$ are the weight matrices. $\overrightarrow{\$R}$, $\overrightarrow{\$Z}$, and $\overrightarrow{R_T}$ are the bias vectors. The final state $\overrightarrow{h_T}$ is calculated by performing elementwise product $\odot$ between $(1-\overrightarrow{Z_T})$ and (h_{T-1}), and performing elementwise product $\odot$ between $(\overrightarrow{Z_T})$ and $(\overrightarrow{G_T})$. Finally, at time T, BiGRU calculates its final output by using the previous frame at time $(T-1)$ and the upcoming frame at time $(T+1)$ using the below equation:

$$h_T = \overrightarrow{h_{T-1}} \oplus \overleftarrow{h_{T+1}}, \quad (16)$$

where $\oplus$ denotes the elementwise summation for forward and backward vectors.

3) Softmax classifier: Finally, we have used the softmax classifier as the last layer to perform multi-classification. Let us assume that the proposed LSTMVAE-BiGRU-based IDS has the last layer as a softmax function. Following that, this layer is given the input sequence $X = \left(X_1, X_2, \ldots X_T^N\right)$, and the output layer of the network produces a one-hot encoded C-dimensional vector y while carrying across T timesteps. Hence, the following formula is used to determine the probability that a single input, X, relates to a certain threat category, y [14].

$$p\left(\hat{y}_c = y_c \mid X\right) = \varrho(X)_{y_c} = \frac{e^{X_c}}{\sum_j^C e^{X_j}} \quad (C = 1, 2, \ldots c) \quad (17)$$

$$L\left(\hat{y}_c, y_c\right) = -\sum_{i=1}^{n}\sum_{c=1}^{C} y_c^{P_i} \cdot \log\left(p\left(\hat{y}_{ic} = y_{ic} \mid P_i\right)\right). \quad (18)$$

The loss during the training procedure is calculated using categorical cross-entropy loss function as given above.

IV. Result Analysis

The proposed approach is implemented using the Tensorflow library and Python 3.5. The experimental setup to perform the simulations are mentioned in Table I. The proposed IDS uses the ToN-IoT dataset that includes multiple types of cybersecurity attacks, including backdoor, DDoS, DoS, injection, MITM, password attack, ransomware, scanning, and XSS. This dataset includes 44 features and the above mentioned labeled attack-types with a normal class. The proposed model tries to solve the multi-class attack detection challenge. The performance of the proposed IDS is evaluated using different evaluation metrics. For instance, we have used False Negative (α), True Negative (β), False Positive (γ), and True Positive (δ) parameters to compute different metrics: Precision (PR) $= \frac{\delta}{\delta+\gamma}$, Detection Rate (DR) $= \frac{\delta}{\alpha+\delta}$, Accuracy (AC) $= \frac{\delta+\beta}{\gamma+\beta+\delta+\alpha}$, False Alarm Rate (FAR) $= \frac{\gamma}{\gamma+\beta}$, and F1 Score $= 2*\frac{PR*RC}{PR+RC}$. First, we pre-processed the dataset and followed a series of steps: Imputation of missing values with the mean of that particular row, followed by converting categorical features into numeric data points using the label-encoding technique. Then, the dataset values were normalized between 0 and 1 using the min-max scaler technique. The ToN-IoT dataset is then divided into a 70-30 ratio of training and testing set. Finally, the proposed approach is compared with some baseline techniques, i.e., random forest (Baseline-1), decision trees (Baseline-2), and naive bayes (Baseline-3), that are commonly used in the literature.

TABLE I: Hyper-parameters used for designing LSTMVAE-BiGRU-based IDS.

Settings	Hyperparameters
Input Layer	44 attributes from the ToN-IoT dataset
Encoder	Hidden Layers (HL)= 4; Hidden Nodes (HN)= (44,35,25,10), tanh function
Decoder	HL= 4; HN= (10,25,35,44), 3 Layers use tanh and the last layer uses sigmoid
LSTMVAE Model	loss='binary crossentropy', optimizer='adam', batch size= 50, epochs= 10
Hidden layers	HL= 5; HN= (30,128,64,15,5)
Output layer	a softmax function 1 normal and 9 attacks
BiGRU Model	optimizer='adam', loss='categorical crossentropy', batch size= 50, epochs= 10

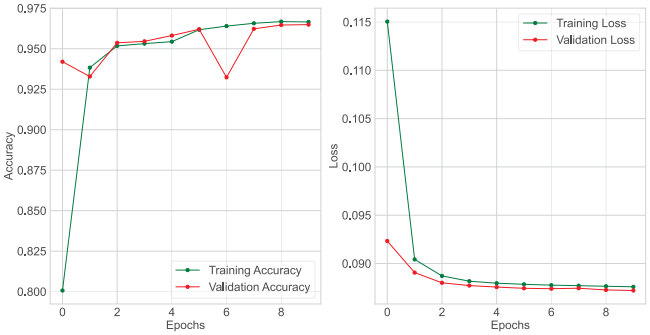

Fig. 2: Accuracy vs. loss for LSTMVAE.

A. Performance Analysis

In this subsection, we analyze the performance of the proposed IDS on various metrics. A detailed explanation is described below.

1) Accuracy vs. loss: The proposed LSTMVAE technique has efficiently learned the dataset. The hyperparameters used to design the LSTMVAE technique are mentioned in Table I. In Fig. 2, we show the outcome in terms of accuracy vs. loss, and the IDS obtained a 0.0872% *validation loss* and 0.9649% *validation accuracy*. The LSTMVAE technique's objective is not to discover these threat observations but to extract impor-

TABLE II: Per-class prediction results (%) for LSTMVAE-BiGRU-based IDS.

Parameters	Backdoor	DDoS	DoS	Injection	MITM	Normal	Password	Ransomware	Scanning	XSS
PR	99.99	94.04	99.18	94.70	94.47	99.99	99.83	99.68	97.70	99.31
DR	99.56	94.67	98.08	96.73	94.76	100.00	99.29	99.79	97.54	98.53
F1	99.77	94.36	98.63	95.71	94.62	99.99	99.55	99.74	97.62	98.92
FAR	0.000007	0.002728	0.000370	0.002496	0.000130	0.000041	0.000075	0.000143	0.001058	0.000309

TABLE III: Comparison of DR with baseline techniques.

Techniques	Backdoor	DDoS	DoS	Injection	MITM	Normal	Password	Ransomware	Scanning	XSS
Baseline-1	99.98	90.40	91.97	93.53	0.00	100.00	97.81	99.40	95.74	85.47
Baseline-2	100.00	100.00	100.00	0.00	0.00	100.00	100.00	100.00	100.00	100.00
Baseline-3	99.22	26.80	91.70	92.96	95.11	100.00	75.32	79.98	96.91	19.02
Proposed IDS	99.56	94.67	98.08	96.73	94.76	100.00	99.29	99.79	97.54	98.53

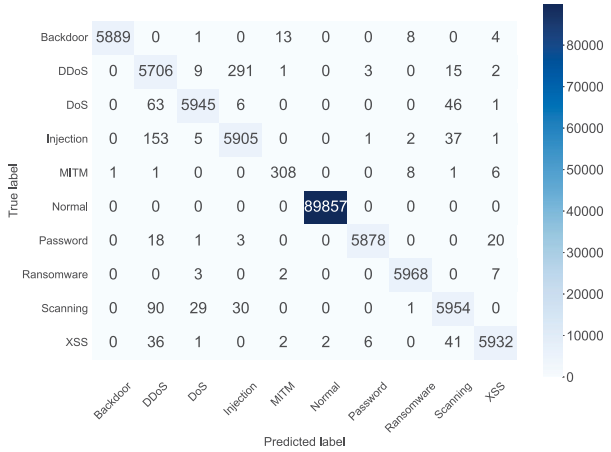

Fig. 3: Confusion matrix for the proposed model.

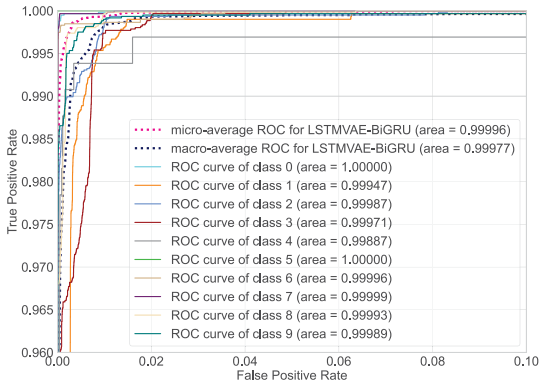

Fig. 4: ROC for the proposed LSTMVAE-BiGRU-based IDS.

tant low-dimensional attributes. Thus, the extracted features are used by the BiGRU with a softmax classifier to detect different attacks present in the dataset.

2) Confusion matrix: Confusion Matrix (CM) is a useful tool that provides details on how well an IDS model performs in binary classifications. All four possible outcomes γ, α, δ, and β are represented in the matrix. It is important to note that positive indicates the IDS expected the traffic record as attack data, whereas negative indicates the IDS anticipated the traffic record as normal data. The diagonal elements in Fig. 3 show the correct classification of various attack vectors and a normal class. It is concluded from the confusion matrix that the false-positive rate for each class in the ToN-IoT dataset is very low and thus the proposed IDS have a high detection rate.

3) Per-class prediction: The per-class prediction analysis in terms of PR, DR, F1 score, and FAR is illustrated in Table II. This analysis is useful when IDS performs a multi-class attack detection process. It can be concluded from Table II that the proposed IDS has achieved PR, DR, and F1 score values between 94%-100%. On the other hand, the false alarm for each class in the ToN-IoT dataset is very low, i.e., 0%. This indicates the success rate of the proposed IDS.

4) ROC Curve: The Receiver Operating Characteristic (ROC) is a useful method for interpreting the results of multi-class vectors in the dataset. As a visual assessment tool, ROC graphs are particularly useful for assessing the efficacy of classifiers. The δ and γ axes are typically extended within the bounds of the two-dimensional ROC space. Based on Fig. 4, the following details are deduced: the reported micro-average Area under the ROC Curve (AUC) for the LSTMVAE-BiGRU-based IDS is significantly high, i.e., 0.99996, and the macro-average AUC value is 0.99977. On the other hand, it can be seen that the AUC values for the different attacks and a normal class are between 0.99887 to 1.00. This indicates that the proposed IDS can easily distinguish between different attacks and a normal class in the ToN-IoT dataset.

B. Comparison with baselines techniques

In this subsection, we compare the performance of the proposed LSTMVAE-BiGRU model against commonly used baseline techniques. First, we compare the performance in terms of per-class DR (multi-class attack detection) scenario. Table III shows the comparison for various attacks and a normal class present in the dataset. It is worth noting that the values for DR metrics are between 94%-100% for LSTMVAE-BiGRU-based IDS. On the other hand, Baseline-2 has achieved higher values for most attacks, but for injection and MITM attacks, the DR is 0%. This table indicates that the proposed IDS has outperformed its competitive models. Finally, we have compared the performance of the proposed IDS with baseline techniques in terms of accuracy, precision, detection rate, and F1 score. Fig. 5 shows this comparison. It can be concluded that the proposed IDS has achieved higher values for these

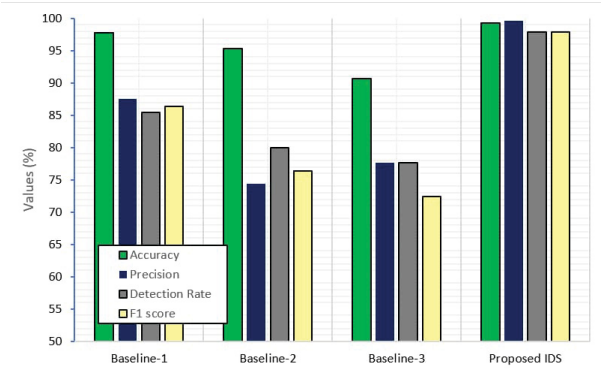

Fig. 5: Performance comparison with baseline techniques.

TABLE IV: Performance comparison with other models.

Works	Year	Method	Data Source	Accuracy
[20]	2021	Classification And Regression Tree	ToN-IoT	88.00%
[21]	2022	Federated IDS	ToN-IoT	94.85%
[22]	2022	Random Forest	ToN-IoT	98.07%
[9]	2023	Convolutional Neural Networks	ToN-IoT	90.55%
Proposed IDS	2023	LSTMVAE-BiGRU	ToN-IoT	99.30%

parameters. We have also compared the performance of the proposed IDS with some recent state-of-the-art methods [20], [21], [9], [22] using accuracy metrics. Table IV highlights this comparison, and we may conclude that LSTMVAE-BiGRU-based IDS performs better in recognizing the cyber-attacks for the IoV networks.

V. Conclusion

To enhance the IoV security, we have proposed a novel IDS based on a generative hybrid deep learning model. The proposed model is designed by combining LSTMVAE, Bi-GRU, and a softmax classifier. The LSTMVAE is used as a statistical feature extraction technique that can learn time series and multivariate data from the IoV network. To evaluate the performance, we feed the extracted features to BiGRU and a softmax classifier for identification and classification of potential cyber-attacks in the IoV network. Experimental results based on the ToN-IoT dataset confirm the superiority of the proposed IDS over commonly used baseline techniques. Future research will include testing the proposed IDS on different datasets.

References

[1] F. Tang, B. Mao, N. Kato, and G. Gui, "Comprehensive Survey on Machine Learning in Vehicular Network: Technology, Applications and Challenges," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 2027–2057, 2021.

[2] M. Placek, "U.s. - connected vehicles 2030," Aug.2021. [Online]. Available: https://www.statista.com/statistics/750113/us-connected-vehicles

[3] Y. Li, Y. Zuo, H. Song, and Z. Lv, "Deep Learning in Security of Internet of Things," *IEEE Internet of Things Journal*, vol. 9, no. 22, pp. 22 133–22 146, 2021.

[4] A. Anzer and M. Elhadef, "A Multilayer Perceptron-Based Distributed Intrusion Detection System for Internet of Vehicles," in *2018 IEEE 4th international conference on collaboration and internet computing (CIC)*. IEEE, 2018, pp. 438–445.

[5] H. H. R. Sherazi, R. Iqbal, F. Ahmad, Z. A. Khan, and M. H. Chaudary, "DDoS attack detection: A key enabler for sustainable communication in internet of vehicles," *Sustainable Computing: Informatics and Systems*, vol. 23, pp. 13–20, 2019.

[6] S. Anbalagan, A. K. Bashir, G. Raja, P. Dhanasekaran, G. Vijayaraghavan, U. Tariq, and M. Guizani, "Machine-Learning-Based Efficient and Secure RSU Placement Mechanism for Software-Defined-IoV," *IEEE Internet of Things Journal*, vol. 8, no. 18, pp. 13 950–13 957, 2021.

[7] L. Yang, A. Shami, G. Stevens, and S. De Rusett, "LCCDE: A Decision-Based Ensemble Framework for Intrusion Detection in the Internet of Vehicles," in *GLOBECOM 2022-2022 IEEE Global Communications Conference*. IEEE, 2022, pp. 3545–3550.

[8] T. Alladi, V. Kohli, V. Chamola, F. R. Yu, and M. Guizani, "Artificial Intelligence (AI)-Empowered Intrusion Detection Architecture for the Internet of Vehicles," *IEEE Wireless Communications*, vol. 28, no. 3, pp. 144–149, 2021.

[9] A. Oseni, N. Moustafa, G. Creech, N. Sohrabi, A. Strelzoff, Z. Tari, and I. Linkov, "An Explainable Deep Learning Framework for Resilient Intrusion Detection in IoT-Enabled Transportation Networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 1, pp. 1000–1014, 2023.

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

[11] I. Ahmed, G. Jeon, and A. Ahmad, "Deep Learning-Based Intrusion Detection System for Internet of Vehicles," *IEEE Consumer Electronics Magazine*, vol. 12, no. 1, pp. 117–123, 2021.

[12] S. Almutlaq, A. Derhab, M. M. Hassan, and K. Kaur, "Two-Stage Intrusion Detection System in Intelligent Transportation Systems Using Rule Extraction Methods from Deep Neural Networks," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[13] J. Ashraf, A. D. Bakhshi, N. Moustafa, H. Khurshid, A. Javed, and A. Beheshti, "Novel Deep Learning-Enabled LSTM Autoencoder Architecture for Discovering Anomalous Events From Intelligent Transportation Systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4507–4518, 2020.

[14] M. Al-Hawawreh, N. Moustafa, S. Garg, and M. S. Hossain, "Deep Learning-Enabled Threat Intelligence Scheme in the Internet of Things Networks," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 4, pp. 2968–2981, 2020.

[15] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," *Sustainable Cities and Society*, vol. 72, p. 102994, 2021.

[16] R. Al-Hmouz, W. Pedrycz, A. Balamash, and A. Morfeq, "Logic-Oriented Autoencoders and Granular Logic Autoencoders: Developing Interpretable Data Representation," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 3, pp. 869–877, 2022.

[17] J. Chen, L. Du, and L. Liao, "Discriminative Mixture Variational Autoencoder for Semisupervised Classification," *IEEE Transactions on Cybernetics*, vol. 52, no. 5, pp. 3032–3046, 2022.

[18] X. Shu, L. Zhang, Y. Sun, and J. Tang, "Host–Parasite: Graph LSTM-in-LSTM for Group Activity Recognition," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 2, pp. 663–674, 2021.

[19] Y. Deng, L. Wang, H. Jia, X. Tong, and F. Li, "A Sequence-to-Sequence Deep Learning Architecture Based on Bidirectional GRU for Type Recognition and Time Location of Combined Power Quality Disturbance," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 8, pp. 4481–4493, 2019.

[20] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and A. Anwar, "TON_IoT Telemetry Dataset: A New Generation Dataset of IoT and IIoT for Data-Driven Intrusion Detection Systems," *IEEE Access*, vol. 8, pp. 165 130–165 150, 2020.

[21] M. Abdel-Basset, N. Moustafa, H. Hawash, I. Razzak, K. M. Sallam, and O. M. Elkomy, "Federated Intrusion Detection in Blockchain-Based Smart Transportation Systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 2523–2537, 2022.

[22] T. M. Booij, I. Chiscop, E. Meeuwissen, N. Moustafa, and F. T. H. d. Hartog, "TON_IoT: The Role of Heterogeneity and the Need for Standardization of Features and Attack Types in IoT Network Intrusion Data Sets," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 485–496, 2022.

BIBLIOGRAPHY

Abdel-Basset, M., Moustafa, N., Hawash, H., Razzak, I., Sallam, K. M. & Elkomy, O. M. (2022). Federated Intrusion Detection in Blockchain-Based Smart Transportation Systems. *IEEE Transactions on Intelligent Transportation Systems*, 23(3), 2523-2537. doi: 10.1109/TITS.2021.3119968.

Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J. & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Transactions on Emerging Telecommunications Technologies*, 32(1), e4150.

Ahmed, I., Jeon, G. & Ahmad, A. (2021). Deep Learning-Based Intrusion Detection System for Internet of Vehicles. *IEEE Consumer Electronics Magazine*, 12(1), 117–123.

Al-Hawawreh, M., Moustafa, N., Garg, S. & Hossain, M. S. (2020). Deep Learning-Enabled Threat Intelligence Scheme in the Internet of Things Networks. *IEEE Transactions on Network Science and Engineering*, 8(4), 2968–2981.

Al-Hmouz, R., Pedrycz, W., Balamash, A. & Morfeq, A. (2022). Logic-Oriented Autoencoders and Granular Logic Autoencoders: Developing Interpretable Data Representation. *IEEE Transactions on Fuzzy Systems*, 30(3), 869-877. doi: 10.1109/TFUZZ.2020.3043659.

Alferaidi, A., Yadav, K., Alharbi, Y., Razmjooy, N., Viriyasitavat, W., Gulati, K., Kautish, S., Dhiman, G. et al. (2022). Distributed deep CNN-LSTM model for intrusion detection method in IoT-based vehicles. *Mathematical Problems in Engineering*, 2022.

Alladi, T., Kohli, V., Chamola, V., Yu, F. R. & Guizani, M. (2021). Artificial Intelligence (AI)-Empowered Intrusion Detection Architecture for the Internet of Vehicles. *IEEE Wireless Communications*, 28(3), 144–149.

Almutlaq, S., Derhab, A., Hassan, M. M. & Kaur, K. (2022). Two-Stage Intrusion Detection System in Intelligent Transportation Systems Using Rule Extraction Methods from Deep Neural Networks. *IEEE Transactions on Intelligent Transportation Systems*.

Alsaedi, A., Moustafa, N., Tari, Z., Mahmood, A. & Anwar, A. (2020). TON_IoT Telemetry Dataset: A New Generation Dataset of IoT and IIoT for Data-Driven Intrusion Detection Systems. *IEEE Access*, 8, 165130-165150. doi: 10.1109/ACCESS.2020.3022862.

Anbalagan, S., Bashir, A. K., Raja, G., Dhanasekaran, P., Vijayaraghavan, G., Tariq, U. & Guizani, M. (2021). Machine-Learning-Based Efficient and Secure RSU Placement Mechanism for Software-Defined-IoV. *IEEE Internet of Things Journal*, 8(18), 13950–13957.

Anbalagan, S., Raja, G., Gurumoorthy, S., Suresh, R. D. & Dev, K. (2023). IIDS: Intelligent Intrusion Detection System for Sustainable Development in Autonomous Vehicles. *IEEE Transactions on Intelligent Transportation Systems*.

Anzer, A. & Elhadef, M. (2018). A Multilayer Perceptron-Based Distributed Intrusion Detection System for Internet of Vehicles. *2018 IEEE 4th international conference on collaboration and internet computing (CIC)*, pp. 438–445.

Ashraf, J., Bakhshi, A. D., Moustafa, N., Khurshid, H., Javed, A. & Beheshti, A. (2020). Novel Deep Learning-Enabled LSTM Autoencoder Architecture for Discovering Anomalous Events From Intelligent Transportation Systems. *IEEE Transactions on Intelligent Transportation Systems*, 22(7), 4507–4518.

Booij, T. M., Chiscop, I., Meeuwissen, E., Moustafa, N. & Hartog, F. T. H. d. (2022). TON_IoT: The Role of Heterogeneity and the Need for Standardization of Features and Attack Types in IoT Network Intrusion Data Sets. *IEEE Internet of Things Journal*, 9(1), 485-496. doi: 10.1109/JIOT.2021.3085194.

Boualouache, A. & Engel, T. (2023). A survey on machine learning-based misbehavior detection systems for 5g and beyond vehicular networks. *IEEE Communications Surveys & Tutorials*.

Chen, J., Du, L. & Liao, L. (2022). Discriminative Mixture Variational Autoencoder for Semisupervised Classification. *IEEE Transactions on Cybernetics*, 52(5), 3032-3046. doi: 10.1109/TCYB.2020.3023019.

de Araujo-Filho, P. F., Kaddoum, G., Campelo, D. R., Santos, A. G., Macêdo, D. & Zanchettin, C. (2020). Intrusion detection for cyber–physical systems using generative adversarial networks in fog environment. *IEEE Internet of Things Journal*, 8(8), 6247–6256.

De Araujo-Filho, P. F., Pinheiro, A. J., Kaddoum, G., Campelo, D. R. & Soares, F. L. (2021). An efficient intrusion prevention system for CAN: Hindering cyber-attacks with a low-cost platform. *IEEE Access*, 9, 166855–166869.

Deng, Y., Wang, L., Jia, H., Tong, X. & Li, F. (2019). A Sequence-to-Sequence Deep Learning Architecture Based on Bidirectional GRU for Type Recognition and Time Location of Combined Power Quality Disturbance. *IEEE Transactions on Industrial Informatics*, 15(8), 4481–4493.

Garg, S., Kaur, K., Kumar, N., Kaddoum, G., Zomaya, A. Y. & Ranjan, R. (2019). A hybrid deep learning-based model for anomaly detection in cloud datacenter networks. *IEEE Transactions on Network and Service Management*, 16(3), 924–935.

Garg, S., Kaur, K., Kaddoum, G., Garigipati, P. & Aujla, G. S. (2021). Security in IoT-driven mobile edge computing: New paradigms, challenges, and opportunities. *IEEE Network*, 35(5), 298–305.

Greenberg, A. (2016). Wired [Format]. Retrieved from: https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/.

Hachimi, M., Kaddoum, G., Gagnon, G. & Illy, P. (2020). Multi-stage jamming attacks detection using deep learning combined with kernelized support vector machine in 5g cloud radio access networks. *2020 international symposium on networks, computers and communications (ISNCC)*, pp. 1–5.

He, K., Kim, D. D. & Asghar, M. R. (2023). Adversarial Machine Learning for Network Intrusion Detection Systems: A Comprehensive Survey. *IEEE Communications Surveys & Tutorials*.

Huddleston, T. (2019). CNBC [Format]. Retrieved from: https://www.cnbc.com/2019/04/03/chinese-hackers-tricked-teslas-autopilot-into-switching-lanes.html/.

Illy, P., Kaddoum, G., Moreira, C. M., Kaur, K. & Garg, S. (2019). Securing fog-to-things environment using intrusion detection system based on ensemble learning. *2019 IEEE wireless communications and networking conference (WCNC)*, pp. 1–7.

Illy, P., Kaddoum, G., de Araujo-Filho, P. F., Kaur, K. & Garg, S. (2022). A hybrid multistage DNN-based collaborative IDPS for high-risk smart factory networks. *IEEE Transactions on Network and Service Management*, 19(4), 4273–4283.

Kaiwartya, O., Abdullah, A. H., Cao, Y., Altameem, A., Prasad, M., Lin, C.-T. & Liu, X. (2016). Internet of vehicles: Motivation, layered architecture, network model, challenges, and future aspects. *IEEE access*, 4, 5356–5373.

Kasongo, S. M. (2023). A deep learning technique for intrusion detection system using a Recurrent Neural Networks based framework. *Computer Communications*, 199, 113–125.

Khraisat, A., Gondal, I., Vamplew, P. & Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. *Cybersecurity*, 2(1), 1–22.

Kostadinov, S. (2017). Understanding GRU Networks [Format]. Retrieved from: https://towardsdatascience.com/understanding-gru-networks-2ef37df6c9be.

Li, Y., Zuo, Y., Song, H. & Lv, Z. (2021). Deep learning in security of internet of things. *IEEE Internet of Things Journal*, 9(22), 22133–22146.

Magaia, N., Gomes, P., Silva, L., Sousa, B., Mavromoustakis, C. X. & Mastorakis, G. (2020). Development of mobile IoT solutions: approaches, architectures, and methodologies. *IEEE Internet of Things Journal*, 8(22), 16452–16472.

Man, D., Zeng, F., Lv, J., Xuan, S., Yang, W. & Guizani, M. (2021). AI-based Intrusion Detection for Intelligence Internet of Vehicles. *IEEE Consumer Electronics Magazine*, 12(1), 109–116.

Miranda, C., Kaddoum, G., Bou-Harb, E., Garg, S. & Kaur, K. (2020). A collaborative security framework for software-defined wireless sensor networks. *IEEE Transactions on Information Forensics and Security*, 15, 2602–2615.

Miranda, C., Kaddoum, G., Boukhtouta, A., Madi, T. & Alameddine, H. A. (2022). Intrusion prevention scheme against rank attacks for software-defined low power IoT networks. *IEEE Access*, 10, 129970–129984.

Moustafa, N. (2021). A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets. *Sustainable Cities and Society*, 72, 102994.

Nie, L., Ning, Z., Wang, X., Hu, X., Cheng, J. & Li, Y. (2020). Data-Driven Intrusion Detection for Intelligent Internet of Vehicles: A Deep Convolutional Neural Network-Based Method. *IEEE Transactions on Network Science and Engineering*, 7(4), 2219–2230.

Olah, C. (2015). Understanding LSTM networks.

Oseni, A., Moustafa, N., Creech, G., Sohrabi, N., Strelzoff, A., Tari, Z. & Linkov, I. (2022). An Explainable Deep Learning Framework for Resilient Intrusion Detection in IoT-Enabled Transportation Networks. *IEEE Transactions on Intelligent Transportation Systems*.

Oseni, A., Moustafa, N., Creech, G., Sohrabi, N., Strelzoff, A., Tari, Z. & Linkov, I. (2023). An Explainable Deep Learning Framework for Resilient Intrusion Detection in IoT-Enabled Transportation Networks. *IEEE Transactions on Intelligent Transportation Systems*, 24(1), 1000-1014. doi: 10.1109/TITS.2022.3188671.

Park, C., Lee, J., Kim, Y., Park, J.-G., Kim, H. & Hong, D. (2023). An Enhanced AI-Based Network Intrusion Detection System Using Generative Adversarial Networks. *IEEE Internet of Things Journal*, 10(3), 2330-2345. doi: 10.1109/JIOT.2022.3211346.

Placek, M. (Aug.2021). U.S. - connected vehicles 2030. Retrieved from: https://www.statista.com/statistics/750113/us-connected-vehicles.

Placek, M. (2021). Statista [Format]. Retrieved from: https://www.statista.com/statistics/1230664/projected-number-autonomous-cars-worldwide/.

Pradhan, M., Mohanty, S. & Seemona, A. O. (2022). Machine Learning-Based Intrusion Detection System for the Internet of Vehicles. *2022 5th International Conference on Computational Intelligence and Networks (CINE)*, pp. 1–6.

Rani, S. J., Ioannou, I., Nagaradjane, P., Christophorou, C., Vassiliou, V., Yarramsetti, H., Shridhar, S., Balaji, L. M. & Pitsillides, A. (2023). A Novel Deep Hierarchical Machine Learning Approach for Identification of Known and Unknown Multiple Security Attacks in a D2D Communications Network. *IEEE Access*.

Sharma, P. & Liu, H. (2020). A machine-learning-based data-centric misbehavior detection model for internet of vehicles. *IEEE Internet of Things Journal*, 8(6), 4991–4999.

Sherazi, H. H. R., Iqbal, R., Ahmad, F., Khan, Z. A. & Chaudary, M. H. (2019). DDoS attack detection: A key enabler for sustainable communication in internet of vehicles. *Sustainable Computing: Informatics and Systems*, 23, 13–20.

Shu, X., Zhang, L., Sun, Y. & Tang, J. (2021). Host–Parasite: Graph LSTM-in-LSTM for Group Activity Recognition. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2), 663-674. doi: 10.1109/TNNLS.2020.2978942.

Talpur, A. & Gurusamy, M. (2021). Machine learning for security in vehicular networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 24(1), 346–379.

Tang, F., Mao, B., Kato, N. & Gui, G. (2021). Comprehensive Survey on Machine Learning in Vehicular Network: Technology, Applications and Challenges. *IEEE Communications Surveys & Tutorials*, 23(3), 2027–2057.

Taslimasa, H., Dadkhah, S., Neto, E. C. P., Xiong, P., Ray, S. & Ghorbani, A. A. (2023). Security issues in Internet of Vehicles (IoV): A comprehensive survey. *Internet of Things*, 100809.

Wang, Y., Wang, J., Jiang, J., Xu, S. & Wang, J. (2023). SA-LSTM: A Trajectory Prediction Model for Complex off-road Multi-agent Systems Considering Situation Awareness Based on Risk Field. *IEEE Transactions on Vehicular Technology*, 1-12. doi: 10.1109/TVT.2023.3287227.

Yang, L., Moubayed, A., Hamieh, I. & Shami, A. (2019). Tree-based intelligent intrusion detection system in internet of vehicles. *2019 IEEE global communications conference (GLOBECOM)*, pp. 1–6.

Yang, L., Moubayed, A. & Shami, A. (2021). MTH-IDS: A multitiered hybrid intrusion detection system for internet of vehicles. *IEEE Internet of Things Journal*, 9(1), 616–632.

Yang, L., Shami, A., Stevens, G. & De Rusett, S. (2022). LCCDE: A Decision-Based Ensemble Framework for Intrusion Detection in the Internet of Vehicles. *GLOBECOM 2022-2022 IEEE Global Communications Conference*, pp. 3545–3550.