An Online Polymorphic Attack Detection Model for Cooperative Intelligent Transportation Systems

A Dissertation

Presented in Partial Fulfillment of the Requirements for the

Doctor of Philosophy

with a

Major in Computer Science

in the

College of Graduate Studies at

University of Idaho

by

Sultan Ahmed Almalki

Approved by:

Major Professor: **Frederick Sheldon**, Ph.D.

Committee Members: Terence Soule, Ph.D.; Marshall Ma, Ph.D.; Bander Al-rimy, Ph.D.

Department Administrator: Terence Soule, Ph.D.

December 2022

Abstract

Cooperative Intelligent Transportation Systems (cITSs) represent one of the Internet of Things (IoT) applications whose purpose is to improve road safety and traffic efficiency. Within this system, vehicles can communicate with one another by establishing a Vehicular Ad-Hoc Network (VANET) along the particular road section of interest. Although such connectivity facilitates the exchange of information related to road safety and traffic efficiency, at the same time connectivity puts vehicles at risk of compromise. An attacker could exploit one or more vehicles weaknesses, and use them to share false information causing congestion and/or life-threatening accidents. Several studies have tried to address this issue. Generally, those studies assume that the network topology and/or attack behavior is stationary. This is certainly not realistic, as the cITS is dynamic in nature, and the attackers may have the ability and resources to change their behaviour continuously. Therefore, these assumptions are not suitable and lead to low detection accuracy and high false alarms. To this end, this study proposes a misbehaviour detection model that can cope with the dynamicity of both cITS topology and attack behaviour. The model starts by addressing the issue of missing data using a local-global Fuzzy clustering estimation method. Then, a Proportional Conditional Redundancy Coefficient (PCRC) is used to calculate the values of redundancy and relevancy coefficients in the goal function of the feature selection. This helps to better estimate the discriminative features during the

model training. The selected features were used to train an online deep learning-based model. The model uses a Bi-variate Moving Average (BiMAV) to observe the polymorphic patterns in the attack's behaviour and re-adjust the security parameters accordingly was trained. In comparison to reported studies, the results show that the proposed method achieved improvement compared to the existing techniques we demonstrated in i) Phase 1 an improvement of ranging between 0.8% - 47% across the metrics (Accuracy, F-measure, False Positive Rates, and Detection Rate), ii) Phase 2 an improvement of ranging between 2.0% - 3.1% across the metric (ACC), iii) Phase 3 an improvement of ranging between 2.4% - 42% across the metrics (ACC, F1, FPR, and DR) in a highly dynamic and potentially contested environment. There are many threats where this approach has much better chances of delivering the needed results and we believe is more resilient (e.g., False Data Injection). The proposed model is expected to overcome the limitations of related solutions by detecting attacks that change their behaviour continuously.

Acknowledgements

First and foremost, I want to thank Allah, the Almighty, for the completion of this dissertation. This work would not have been possible without the kind support and help of many individuals. I am extending my sincere thanks to all of them.

I would like to express my deepest gratitude to my Major Professor, Frederick T. Sheldon, for his excellent guidance, caring, patience, and providing me with an excellent atmosphere for doing this research. I am extremely grateful to the rest of my committee: Prof. Terence Soule, Prof. Marshall Ma, and Prof. Bander Al-rimy, for their encouragement and insightful comments.

I would like to thank all my instructors for their hard work and dedication in providing me with a comprehensive and valuable education.

I would like to thank all the staff of the College of Graduate Studies for their help and support throughout my study at UoI. I would like to thank many other friends who made my life at UoI very enjoyable.

I would like to thank and acknowledge my government Kingdom of Saudi Arabia, and Saudi Arabia Culture Mission (SACM) for their support and funding my studies in UoI and finishing this dissertation.

Finally, I am proud to be student in the Department of Computer Science at University of Idaho.

Dedication

I dedicate this doctoral dissertation to my parents (Ahmed Almalki and Fawziah Altouri), my brothers, my sisters, my entire extended family, and the people who have supported me during the challenges of this scientific study.

Contents

Abstract . . . . . **ii**

Acknowledgements . . . . . **iv**

Dedication . . . . . **v**

List of Abbreviations . . . . . **xiv**

1 Introduction **1**

1.1 Cooperative Intelligent Transportation Systems 1

1.1.1 The cyber threats against cITS 4

1.1.2 Existing Solutions 5

1.2 Problem Statement 8

1.3 Research Aim 9

1.4 Questions and Objectives 10

1.5 Research Motivation 11

1.6 Research Contribution 13

1.7 Author's Related Publications 15

1.8 Dissertation organization 15

2 Background and Related works **17**

2.1 Existing Intrusion Detection Solutions in cITS 18
2.2 Misbehaviour Detection Solutions in cITS 23
2.2.1 Data-Centric Misbehaviour Detection Solutions 23
2.2.2 Node-Centric Misbehaviour Detection Solutions 27
2.3 Comparing the IDS versus MDS cITSs Approaches 31
2.4 Conclusion 33

3 Methodology 34
3.1 Phase 1: Data Pre-processing and Feature Extraction 35
3.2 Phase 2: Feature Selection 39
3.3 Phase 3: Model Training/Testing 40
3.4 The Dataset 45
3.5 Experimental Environment Setup 46
3.6 Evaluation Metrics 46
3.7 Summary 47

4 Deep Learning to Improve False Data Injection Attack Detection in Cooperative Intelligent Transportation Systems 49
4.1 Abstract 50
4.2 Introduction 50
4.3 Proposed Methods 53
4.3.1 Data Pre-processing 55
4.3.2 Data Normalization and Standardization 55
4.3.3 Missing values Imputation using Multivariate Fuzzy C - Means . . 56
4.3.4 Multivariate fuzzy c-means for missing data estimation 57

4.4 Results and Discussion 59
4.5 Conclusion 65

5 Disrupting the Cooperative Nature of Intelligent Transportation Systems 66

5.1 Abstract 66
5.2 Introduction 68
5.3 Related Work 70
5.4 Methodology 73
5.5 Results and Discussion 77
5.5.1 Dataset 78
5.5.2 Experimental Results and Analysis 79
5.6 Conclusion 84

6 Adaptive IDS for Cooperative Intelligent Transportation Systems Using Deep Belief Networks 86

6.1 Abstract 86
6.2 Introduction 87
6.3 Related Works 89
6.4 Methodology 93
6.4.1 Training and Testing 95
6.5 Model Adaptation using Bi-Variate Moving Average 96
6.6 The Dataset 97
6.7 Experimental Environment Setup 98
6.7.1 Evaluation Metrics 99
6.8 Experimental Results 99
6.9 Conclusions and Summary 105

7 Summary, Conclusion and Future Work 106

7.1 Future Work 108

7.1.1 Data diversity 108

7.1.2 Attack Diversity 109

7.1.3 Multiple Sources 109

Bibliography 110

List of Figures

3.1 The general architecture of the proposed model. 36
3.2 Data Pre-processing and Feature Extraction architecture. 38
3.3 Feature Selection architecture. 39
3.4 Model Training/Testing architecture. 42

4.1 Accuracy comparison between MFC-DI and Pearson-Based imputation for the LR, SVM, and CNN. 62
4.2 F1 measure comparison between MFC-DI and Pearson-Based imputation for the LR, SVM, and CNN. 62
4.3 FPR comparison between MFC-DI and Pearson-Based imputation for the LR, SVM, and CNN. 63
4.4 DR comparison between MFC-DI and Pearson-Based imputation for the LR, SVM, and CNN. 63

5.1 Pseudo code of our EJMI feature selection technique 77
5.2 Accuracy comparison between the proposed PCRC incorporated into the EJMI for feature selection and the existing feature selection techniques using DNN. 83

5.3 Accuracy comparison between the proposed PCRC incorporated into the EJMI for feature selection and the existing feature selection techniques using SVM. 83

5.4 Accuracy comparison between the proposed PCRC incorporated into the EJMI for feature selection and the existing feature selection techniques using LR. 84

6.1 Comparison of the proposed ADBN-IDS with SVM and LR in terms of detection accuracy. 102

6.2 Comparison of the proposed ADBN-IDS with SVM and LR in terms of detection rate. 103

6.3 Comparison of the proposed ADBN-IDS with SVM and LR in terms of false positive rate. 103

6.4 Comparison of the proposed ADBN-IDS with SVM and LR in terms of F measure. 104

6.5 Area under the curve comparison for several detection thresholds. 104

List of Tables

1.1 Summarizes the problem situation and solution concept. 14

2.1 Existing IDS studies for cITS. 22

2.2 Existing research in MDS for cITS. 29

3.1 Research Plan. 43

4.1 Performance comparison of the LR, SVM, and CNN using the data processed by the MFC-DI. 60

5.1 Accuracy comparison between the proposed PCRC incorporated into the EJMI for feature selection and the existing feature selection techniques using DNN 81

5.2 Accuracy comparison between the proposed PCRC incorporated into the EJMI for feature selection and the existing feature selection techniques using SVM 82

5.3 Accuracy comparison between the proposed PCRC incorporated into the EJMI for feature selection and the existing feature selection techniques using LR 82

6.1 The experimental evaluation results for the proposed ADBN-IDS in terms of accuracy, detection rate, false positive rate, and F measure. 101

6.2 The experimental evaluation results for the proposed SVM-IDS in terms of accuracy, detection rate, false positive rate, and F measure. 101

6.3 The experimental evaluation results for the proposed LR-IDS in terms of accuracy, detection rate, false positive rate, and F measure. 102

7.1 Phase 1 percentage improvement compared to the existing techniques (refer to Section 4.4, Table 4.1)). 107

List of Abbreviations

CITS	Cooperative Intelligent Transportation Systems
IDS	Intrusion detection system
MDS	Misbehaviour Detection System
NGSIM	Next Generation Simulation
PCRC	Proportional Conditional Redundancy Coefficient
MI	Mutual Information
EJMI	Enhanced Joint Mutual Information
VANET	Vehicular Ad-Hoc Networks
DBN	Deep Belief Networks
BiMAV	Bi-variate moving average
ADBN-IDS	Adaptive Deep Belief Network-Based intrusion detection system
X_i	Data value of X
$\overline{X}$	Mean of X
Y_i	Data value of y
$\overline{Y}$	Mean of y

n	Number of data points
K	number of clusters
C	the number of centroids
X_k	Candidate Feature
Y	Class label
β	Marginal Redundancy
γ	Conditional redundancy
X_j	Feature
S	Selected set
F	Size of original set
Y_i	Output values
l	Number of instances
W	windows
σ	Standard deviation
r	Correlation Coefficient
X	Original Value
$X_{\max}$	Maximum Value of X
$X_{\min}$	Minimum Value of X
μ	Sample Mean

t	Threshold
Cov	Sample Covariance
Var	Sample Variance
γ_{ik}^{m}	The membership of j^{th} data point to the k^{th} centroid
d_{ik}	The Euclidean distance
U	Matrix
SVM	Support Vector Machine
LR	Logistic Regression
CNN	Convolutional Neural Network
ACC	Accuracy
F1	F-score or F-measure
FPR	False Positive Rate
DR	Detection Rate
CA-DC-MDS	Context-Aware Data-centric Misbehavior Detection Scheme
HCA-MDS	Hybrid and Multifaceted Context-aware Misbehavior Detection
IoT	Internet of Things
RSUs	Roadside units
V2V	vehicle-to-vehicle
V2I	vehicle-to-infrastructure

ETSI	European telecommunications standardization organization
SAE	Society of Automotive Engineers
CACC	Cooperative Adaptive Cruise Control
DENM	Decentralized Environmental Notification Message
CAM	Cooperative Awareness Message
QoS	Quality of Service
QoE	Quality of Experience
CTI	Collaborative Trust Index
ECD	Entity-Centric detection
DCD	Data-Centric detection
FDI	False data injection
MIFS	Mutual Information based Feature Selection
MRMR	Maximum Relevance Minimum Redundancy
JMI	Joint Mutual Information

Chapter 1

Introduction

In the era of Internet of Things (IoT), many technologies and devices are integrated into a common smart infrastructure whose components could work cooperatively to achieve the different tasks in daily life and business. Among the IoT application is the Cooperative Intelligent Transportation Systems (cITSs) whose purpose is to improve road safety and traffic efficiency [1]. Within a specific road section, vehicles could create Vehicular Ad-Hoc networks (also called VANETs), which the vehicles can use to communicate with each other and exchange different types of data regarding the traffic situation and safety information. However, such connectivity poses many threats to the cITSs nodes, which could come from inside or outside the network.

1.1 Cooperative Intelligent Transportation Systems

The Cooperative Intelligent Transportation Systems (cITS) is one of IoT applications whose purpose is to improve road safety and traffic efficiency [1]. In cITS, the vehicles within a specific range become connected to one network, through which the data and information are exchanged between those vehicles. This has increased the performance, effectiveness and process efficiency and maximized the productivity of these systems. The cITSs are designed to support the autonomous vehicles and improve road safety and

traffic efficiency [1]. To achieve such a goal, several hardware and software components work collaboratively to observe, collect and analyze related data exchanged between the different components of cITS.

Three main components constitute cITS, namely Vehicles, Backend systems and Roadside units (RSUs). The vehicles are equipped with a set of sensors that collect traffic-related data from the surrounding environment, such as velocity, acceleration, GPS, and density. The collected data carry information about road conditions and traffic situations in the vicinity of the vehicle. Backend systems are used to store and analyze the traffic data and send notifications/alerts to vehicles and/or road service providers. RSU is a backbone that connects vehicles on the road section with the backend systems [1]. The purpose of RSUs is to connect the vehicles with some backend systems like traffic control. These components work cooperatively and exchange different types of information. The context information includes, but is not limited to, lane change warnings, accident reporting, and cooperative adaptive cruise control. As such information is relevant to all vehicles in the same vicinity, point-to-point communication is not an efficient way to exchange this information among neighboring vehicles. Therefore, cITS utilizes the broadcasting approach to share the context data among different nodes [1].

The cITS system is composed of two communication schemes, namely vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication. These schemes are proposed to follow the IEEE 802.11p band, which sets the rules and standards for MAC and PHY layers for short-range communication between vehicles. In this communication band, networks operate at 5.9 GHz with a communication range between 100 and 500 meters. Such communication setup relies on the scenario and environment in the vicinity. IEEE 802.11p has been adopted world-wide and used as a basis for cITS's applications. A family

of standards for cITS, referred to as ITS-G5, which provides communication primitives, is defined by the European telecommunications standardization organization (ETSI). On top of these standards, safety and entertainment applications are being developed. In the U.S., similar protocols are used in the IEEE 1609 family of standards, in conjunction with the Society of Automotive Engineers (SAE).

Unlike the routing in conventional networks such as the Internet, multicast and broadcast are the networking approaches used in cITS [2, 3]. These approaches are suitable for several applications like lane-change warnings, cooperative adaptive cruise control (CACC) and city-scale traffic flow optimization. CACC is an application that has gained significant attention in recent years and is essentially an extension of adaptive cruise control (ACC), which reduces the safety distance between vehicles. In CACC, vehicles periodically exchange position information to form a very tight formation that would normally be susceptible to collisions. This is a form of partially autonomous driving that goes beyond the potential benefits from sensor-equipped vehicles. The data exchanged between the network participants, vehicles and infrastructure alike, are similar in that they are relevant to all receivers. Therefore, addressing packets to specific vehicles does not make sense; instead, addressing refers to the local neighbourhood (1-hop broadcast), specific regions (geocast) and infrastructure. This style of addressing allows the network to exploit the simple fact that wireless networks are a broadcast medium by nature.

The cITS utilizes one of two standards as an information-sharing mechanism, namely the European standard [4] and the American standard [4] . The cITS context information in the European standard consists of two messages, the Cooperative Awareness Message (CAM) and the Decentralized Environmental Notification Message (DENM). While CAMs are sent periodically, DENMs are event-driven and are only sent when an event has

occurred. The CAM consists of information about the vehicles like position, size, speed and steering wheel angle. In contrast, DENM contains information about a certain event like lane changing and sudden braking. On the other hand, cITS context information in the American standard combines CAM and DENM into Basic Safety Message (BSM). BSM will be used when discussing the combination of CAM and DENM messages. The first part of BSM, as well as CAM in the European standard, carries information about position, heading, speed, acceleration, steering wheel angle, vehicle role, vehicle size and status of vehicle light [1]. Unlike the first part of BSM that is included in all BSM messages, the second part of BSM (which corresponds to DENM in the European standard) is included only when an event happens.

However, the connectivity provided in cITSs comes at the cost of several threats. These threats could be categorized into threats against system and threats against data [5]. While the former tries to disable or disrupt the function of one or more components in the vehicle's navigation system such as On-Board Unit, the latter tries to corrupt, falsify, and/or manipulate the mobility data exchanged between the neighboring vehicles. These threats could come in the form of malware attacks or human-crafted and organized attacks.

1.1.1 The cyber threats against cITS

Many of the attacks that target smart vehicles come from adversaries who use sophisticated strategies to carry out sustainable attacks like malware and botnets [2, 3]. Such attacks could originate from outside or inside of the network. While attacks from outside can be easily detected and thwarted at the perimeters of the network, insider attacks come from inside of the network and are usually carried out by legitimate, yet compromised vehicles. Based on the attacker location, threats against cITS could be categorized as intrusion

attacks and misbehaving attacks.

Intrusion attacks are mainly launched by intruders from outside of the network such as in replay attack, jamming attack, Sybil attack, and false data injection attacks. In jamming attack, the attacker sends a burst of messages to a specific vehicle, causing a disruption in the communication between the targeted vehicle and other vehicles in the network. Replay attacks intercept the messages exchanged between the vehicles and later re-transmit them for the purpose of impersonation and/or stealing the identity. In Sybil attack, the attacker uses multiple identities in order to deceive other vehicles by reporting a fake road congestion. False data injection is another type of attack, in which the attacker sends false information about the current traffic situation on the road for the purpose of disrupting the traffic or triggering a congestion.

Misbehavior attacks, on the other hand, are launched from the inside of the network by hijacking a legitimate vehicle and using it to manipulate and share false information among the neighboring vehicles. These threats include data manipulation, falsification, and corruption [6]. That is, BSM of the compromised vehicle can be manipulated by attackers to include false information and share it with the neighboring vehicles [7]. Such false information might trigger severe reactions like sudden braking, lane changing, and speed-limit exceeding, which could lead to life-threatening situations. Therefore, protecting BSM messages against the misbehaving attacks is crucial to ensure cITS security and road safety. The misbehavior detection in cITS ecosystems is categorized into node-centric detection and data-centric detection [1].

1.1.2 Existing Solutions

Several studies have been devoted to counteracting attacks in cITS. These solutions are categorized into intrusion detection and misbehavior detection. The intrusion detection

solutions focus on protecting the network against the attacks launched from the outside of the network. IDSs look for patterns related to known attacks like Sybil, Malware, and Dos attacks and raise an alert when matching is found. They can also compare the incoming patterns with the patterns of normal applications and raise an alert when the matching is not found. Unlike IDSs, the purpose of misbehavior detection systems (MDSs) in cITS is to detect attacks launched from inside the network. The problem with insider attacks is that the attacker uses legitimate yet compromised nodes to launch a chain of attacks against the network, which makes it less suspicious to traditional intrusion detection solutions [8]. Even though several solutions have been proposed to mitigate these attacks, some underlying assumptions like stationary context for designing these solutions are not suitable given the highly dynamic and ephemeral nature of cITS. Building data-driven detection models on such stationary assumption renders these solutions unaware of the change in the driving situation as well as network topology in the road section. Therefore, these solutions become out of date quickly. The MDSs are categorized into node-centric and data-centric.

The node-centric MDSs determine the maliciousness of a vehicle based on its behavior on the road section. Such behavior is also used to determine the degree of trustworthiness of legitimate vehicles. The behavior of a node in the cITS could be perceived by observing the number of messages sent by the vehicle and the validity of the format of these messages. A reputation value is calculated to determine the trustworthiness of each node in the cITS. Such calculation could use the voting to identify whether a vehicle is misbehaving. However, relying on the behavioral aspect of the nodes for building security measures is sub-optimal given the non-stationary nature of the cITS environment, wherein the nodes change their behavior according to the topological changes in the network. In

addition, the voting approach used to determine the trustworthiness of a node relies on an underlying premise that the majority of nodes are honest, and the attackers target only a limited number of vehicles in the cITS. Such assumption does not hold with the advanced strategies and techniques employed by the attackers like malware and botnets, which could compromise most of the nodes and consequently create a majority dishonest. Furthermore, the reputation mechanism used by node-centric solutions is susceptible to sudden misbehaving or faulty vehicles [1].

The data-centric misbehavior detection observes the data and messages exchanged between the participating vehicles and performs several checks to find out whether they are falsified, manipulated, and/or tampered with. Particularly, the data pertaining to safety and traffic efficiency exchanged among the neighboring vehicles within the cITS are vetted against several criteria like consistency and plausibility to determine whether they are trustworthy. These data are contained within a type of message called Basic Safety Messages (BSMs). These messages carry two types of information, namely Cooperative Awareness Messages (CAMs) and the Decentralized Environmental Notification Messages (DENMs). The BSMs contain the contextual information including, but not limited to position, heading, speed, acceleration, steering wheel angle, vehicle role, vehicle size and status of vehicle's light [1]. Although exchanging the BSM data facilitates the communication between the neighboring smart vehicles, they could be manipulated and/or falsified either by attackers or compromised nodes, which pose serious threats to road safety or causes congestion on the road section. To avoid such situations, the data-centric MDSs are used to vet the BSM messages to determine whether they are consistent with the general context along the road section. The vetting also helps to identify the plausibility of the data contained within these BSM messages and whether

they are in-line with the data coming from other sources (nodes) in the cITS system.

As pointed out previously, both node-centric and data-centric approaches employed by MDSs for the cITS systems try to assess the trustworthiness of the nodes and data they share with each other. These approaches rely on historical data to evaluate the new instances and/or data. However, both approaches are not suitable for highly dynamic environments like cITSs wherein the nodes join and leave the network very quickly, causing a rapid change in the network topology that makes it difficult to capture holistic patterns that cover all behavioral aspects. Thus, relying on static security solutions with rigid thresholds is not suitable as they could become outdated and, consequently, increase the rate of false alarms and decrease the detection accuracy. Although some solutions have been proposed to overcome this issue by building adaptive IDSs that could cope with the dynamicity of the cITS systems, they lack sufficient data that represent the new situation immediately after the change in the topology is detected. Consequently, these newly formed models suffer from low accuracy and high false alarms.

1.2 Problem Statement

Several data-centric misbehavior detection models utilize mobility data like Basic Safety Messages (BSM) exchanged within cITS to classify vehicles as benign or malicious [2, 3, 9–11]. These data are vetted using several metrics like plausibility and consistency checks to evaluate the trustworthiness of the participating vehicles, which facilitates identification of the misbehaving vehicles in cITS network. The plausibility and consistency checks rely on multiple thresholds that separate the normal from abnormal events. However, these solutions overlook the non-stationary nature of the cITS systems, which invalidates the underlying knowledgebase of these models very quickly.

Although [3] addressed this issue by incorporating an adaptation mechanism to the

detection model to adjust the security thresholds dynamically in real-time, they assume that the data related to the new driving situation are sufficient. However, this assumption does not hold as the amount of data collected at early stages after cITS's topology change might not be sufficient to accurately determine and build the new thresholds of the security profiles. Concretely, such early data are notoriously sparse because they contain lots of missing (null) as well as immature values, which have a negative impact on the ability of detection model to extract features from the data. Such data sparsity and immaturity obstruct the ability of detection models to estimate features significance accurately. Consequently, existing solutions are unable to select between significant features extracted accurately. Furthermore, the adaptation mechanism proposed by the existing misbehavior detection solutions is limited to the dynamic nature of the cITS systems and overlooks the dynamicity of the attacks. Consequently, it becomes difficult for such solutions to identify polymorphic attacks that use sophisticated approaches to change the behavior at each round to deceive the existing measures. To this end, the present study will address these issues by proposing a detection model that can detect evasive and polymorphic attacks which increase the detection performance.

1.3 Research Aim

The aim of this research is to propose and develop an online context-ware misbehavior detection model able to detect the sophisticated and evasive attacks accurately using deep learning along with features significance estimation and selection techniques. The proposed model will overcome data insufficiency during the initial stages of the online model's operation.

1.4 Questions and Objectives

The following are the research questions that will be addressed:

1. How can statistical and clustering techniques be used to compensate the mobility data insufficiency and accurately estimate the missing and yet unobserved values during the early stages of model's formation?

2. How can deep learning be used to accurately calculate the significance of features extracted from the incomplete mobility data during the early stages of model's formation?

3. How can an adaptive approach be utilized to cope with the dynamicity and polymorphic nature of attacks?

To address these questions, the objectives of this dissertation are as follows:

1. To propose a local-global clustering-based estimation technique to improve the accuracy of missing values' imputation in the mobility data collected at the early stages of model's formation.

2. To propose an enhanced feature selection technique by incorporating a Proportional Conditional Redundancy Coefficient (PCRC) into the goal function of the joint mutual information feature selection to improve redundancy co-efficient value calculation, which consequently improves the feature significance estimation.

3. To propose a deep learning-based Intrusion detection model by integrating the techniques proposed in (1) and (2) into a deep belief network structure.

1.5 Research Motivation

Vehicles in cITS are vulnerable to many forms of cyberattacks that compromise the data exchanged between the vehicles, causing many operational disruptions like road congestion and accidents [3]. Existing IDS solutions try to protect the cITS systems by introspecting the patterns of the behavioral signatures of the cyberattacks, based on which new attacks are identified. Several MDSs focus on the data and messages exchanged between the neighboring vehicles in cITS [2, 3, 12]. These solutions determine the trustworthiness of the newly received packets using consistency and plausibility criterion. By assessing the trustworthiness of the data shared between cITS nodes, these solutions determine whether a vehicle is malicious. To make such decision, these solutions compare the behavior and/or data of the node in question with those of other nodes in its vicinity. If the node deviates from the other nodes, it is considered anomalous. if it dose not deviate, it is considered benign otherwise.

Among the challenges pertaining to attack detection in the cITS systems is the evasion behavior that attackers employ to deceive the detection solutions. In particular, the attackers always change their behavior to avoid detection. In the context of data falsification, attackers try to randomize the range of values used to replace the original data. To make it even more challenging, they try to use values that overlap with the normal profile so the data would look original. These kind of attacks can be referred to as polymorphism. Such polymorphic attacks are more challenging as they generate new sets of values every time the system is targeted with false data injection. This makes the predefined thresholding-based detection difficult.

To detect the novel, polymorphic attacks in the cITSs, the anomaly detection approach has been employed by several solutions [13–15]. The assumption is that, by profiling

the normal behavior and defining the boundary of such profile, it is easy to identify the abnormal behavior that falls outside these boundaries. Such an abnormal behavior is most probably coming from attackers [15]. To define the normal profile, existing solutions introspect from the data the patterns that represent the normal behavior. Based on the source of the data, these patterns are categorized into context-based, content-based and entity-based. However, the existing anomaly detection solutions for cITSs consider the normal behavior of legitimate nodes as stationary [2] and overlook the possibility that legitimate nodes could behave maliciously. Such an assumption does not hold as a node could be compromised, and the nature of a node will change from benign to malicious as well. Consequently, these solutions are susceptible to the concept drift, which invalidates security thresholds and settings currently applied by the existing models.

The security thresholds could also be invalidated due to the change in attacker's behavior [16]. Existing solutions rely on the premise that the attackers always follow identical or similar attack strategies when invading the cITS systems. Therefore, these solutions build the security measures suitable for countering predefined situations. However, these solutions are not aware of the obfuscation techniques and strategies that sophisticated attackers and malicious software use to deceive the detection [17]. Therefore, these solutions become attack specific and are unable to identify the polymorphic attacks. That is, by employing polymorphic malware and obfuscation techniques, the attackers have the ability to continuously change their strategies and behavior [18]. Consequently, if security measures are not aware of the dynamic nature of the attack behavior and strategies, they will become outdated very quickly and unable to cope with the new attack strategies.

In light of the abovementioned limitations, it is imperative to build more accurate and adaptive IDS solutions that can cope with the dynamic nature of both cITS environment

and attack strategies. In addition, any proposed solution needs to identify the attacks that try to create a majority of compromised nodes by employing malware and botnets to carry out massive attacks. As such, any proposed solution should be robust enough to identify the false data shared among the vehicles in cITS, even in a situation when most of the nodes are compromised.

1.6 Research Contribution

The problem that this study will address is the inaccuracy of existing IDSs when identifying the attacks launched internally against cITS nodes, which is based on the unrealistic assumption that the cITS network is stationary. Although some studies employed adaptive mechanisms that can cope with the changes in the situation, the data collected immediately after the topology changes is limited and might lack sufficient patterns that serve as evidence for the misbehaving attacks. The early data contain many missing and immature values, which worsen when data dimensionality is high. Imputing the missing values based on immature early data is challenging. Such data insufficiency also adversely affects the ability of detection model to estimate features' significance accurately. Furthermore, these solutions are built based on the premise that the attack is stationary, which does not hold as attackers can change their behaviour to deceive the IDS.

To this end, this research addresses the aforementioned issues by providing an accurate IDS solution for cITS systems. The aim of the proposed solution is to estimate the missing values using a local-global estimation method. This method will utilize the known data in the local attribute (univariate) as well as the corresponding data of other attributes (multivariate), especially those in high correlation with that attribute. Contrary to existing imputation techniques that calculate the missing value based on the values in the same attribute, the proposed solution will also involve the values of other attributes

in the calculation. As the significance of the respective values from the correlated attributes varies, the correlated values will be weighted based on the degree of correlation. The expectation is, by having strongly correlated attributes, the distribution of values in one attribute will follow the distribution of the value in the counterpart attribute, which facilitates the calculation of the missing values. Additionally, the proposed model integrates a PCRC for features significance estimation. The role of deep learning is to determine the optimal values of redundancy and relevancy coefficients in the goal function of the feature selection technique. These coefficients conflict most of the time and need to be chosen prudently. By involving the deep learning in redundancy-relevancy coefficients calculation, the intrinsic characteristics of the data can be perceived more clearly, which compensates for the data insufficiency at the early stages of the formation of online adaptive models. Table 1.1 summarizes the problem situation and solution concept.

Table 1.1: Summarizes the problem situation and solution concept.

Problem Description	**Research Contribution**
Data collected immediately after the topology changes is limited and might lack sufficient patterns with many missing and immature values, which makes it challenging to accurately impute those missing values.	A local-global Fuzzy clustering estimation method is used to estimate the missing data in the local feature.
Insufficient data negatively affects the ability of detection model to estimate features significance accurately.	A Proportional Conditional Redundancy Coefficient (PCRC) is used to calculate the values of redundancy and relevancy coefficients in the goal function of the feature selection.
Existing IDS solutions for cITS assume that the attack behaviour is static (stationary), which renders these solutions outdated quickly.	A deep learning approach that uses a Bi-variate moving average (BiMAV) to observe the polymorphic patterns in the attack's behaviour and re-adjust the security parameters accordingly.

1.7 Author's Related Publications

1. Almalki, S. A., Song, J. (2020). A review on data falsification-based attacks in cooperative intelligent transportation systems. International Journal of Computer Science and Security (IJCSS), 14(2), 22.

2. Almalki, S. A., Sheldon, F. T. (2021, October). Deep Learning to Improve False Data Injection Attack Detection in Cooperative Intelligent Transportation Systems. In 2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON) (pp. 1016-1021). IEEE.

3. Almalki, S. A., Abdel–Rahim, A., Sheldon, F. T. (2022, June). Disrupting the Cooperative Nature of Intelligent Transportation Systems. In 2022 IEEE World AI IoT Congress (AIIoT) (pp. 131-137). IEEE.

4. Almalki, S. A., Abdel–Rahim, A., Sheldon, F. T. (2022, June). Disrupting the Cooperative Nature of Intelligent Transportation Systems. In 2022 IEEE World AI IoT Congress (AIIoT) (pp. 131-137). IEEE.

1.8 Dissertation organization

In this chapter, the problem formalization was discussed, including the research questions and objectives. The rest of this dissertation is organized as follows. Chapter 2 explores the literature related to the IDS for cITS ecosystems. A comprehensive and thorough investigation and analysis of the state-of-the-art solutions are provided. It also summarizes the current research issues and directions. Then, the research methodology along with the research framework, plan, dataset, and evaluation metrics are detailed in Chapter 3. In Chapter 4, the design and development of the fuzzy c-means method for missing data

estimation was discussed. Chapter 5 presents the design and implementation of the PCRC method for an improved joint mutual information features selection technique. In Chapter 6, the design and implementation of the adaptive IDS model using the Bi-variate Moving Average methods is elaborated. This dissertation concludes with a summarization and analysis of the research findings, contributions, and implications and provides suggestions for the future work in Chapter 7.

Chapter 2

Background and Related works

Cooperative Intelligent Transportation System (cITS) is one of IoT applications whose purpose is to enhance drive safety and efficiency. cITS has several components including vehicles, roadside units and backend systems. Like many IoT applications and systems, cITSs are susceptible to a wide range of intruding or misbehaving attacks that could be launched by attackers from inside or outside of the network. When connected to one network, vehicles in cITS become vulnerable to many threats that compromise the confidentiality, integrity, and availability of the data that they exchange [3]. Attackers could compromise such nodes and exploit and manipulate the information they share with other nodes, causing severe complications such as traffic congestions and accidents. Once a vehicle is compromised, it can also be used to launch several types of attacks against other vehicles and/or components of cITS. Such attacks impede the momentum of the integration of cITS technology with existing infrastructure.

To safeguard cITS, the security solutions need to detect the attacks first. As such, it is imperative that detection solutions notify the defense system about the presence of attack. To detect intrusion and misbehavior attacks against cITS, several solutions have been proposed, most of which are data-driven and rely on different types of data collected during the normal operations and/or attacks to build the detection models. Using such

data, several statistical, machine learning and artificial intelligence techniques have been utilized to model the normal and attack profiles and calculate the detection thresholds and parameters. In the following subsections, the studies related to intrusion detection as well as misbehavior detection in cITSs are discussed.

2.1 Existing Intrusion Detection Solutions in cITS

As pointed out previously, IDS focus on preventing the attacks launched by attackers from outside of the participating vehicles. These solutions try to identify specific types of attacks such as jamming, replay, and sybil attacks. They are normally applied either globally at the main location in the cITS system like RSUs or locally on the vehicle's level. Like IDSs that work on traditional networks, intrusion detection system on cITSs can work cooperatively such that vehicles can share the knowledge about new and emerging threats among each other.

M. Aloqaily et al. [19] proposed a cloud-based IDS for smart vehicles that guarantees user's Quality of Service (QoS) and Quality of Experience (QoE). The vehicles are grouped into different clusters, and the vehicles in each cluster are connected to a cluster head whose purpose is to communicate with Trusted Third-Party entities, which act as mediators between service requesters and providers. The proposed IDS has three phases, traffic analysis, data reduction, and classification. In the traffic analysis phase, the collected data are analyzed to identify the behavioral patterns and features. Irrelevant and insignificant features are then removed during the data reduction phase to reduce data dimensionality and prevent overfitting. The model is then built by training a deep learning classifier using the selected features. Deep Belief Networks is employed for data reduction and the Decision Tree for the classification. Dividing vehicles in a road section into clusters each with a cluster head increases the network homogeneity, which facilitates the identification

of anomalous events and entities. However, dividing smart vehicles into clusters and appointing a cluster head for each cluster is challenging given the ephemeral nature of cITS networks. That is, cITS networks are highly dynamic, which renders the clustering approach ineffective as the rate of joining and leaving a particular cluster is very high. Moreover, the cluster head, as with any other vehicle in the cluster, has a short lifetime within the cluster, which makes it unreliable as a mediator between the cluster's vehicles and service providers.

To cope with the highly dynamic nature of cITS, a Trust-aware Collaborative Learning Automata-based IDS was proposed by [20]. The model integrates a Collaborative Trust Index (CTI) into a classification algorithm in order to cover as many types of attacks that target smart cITSs as possible. The CTI is a trustworthiness evaluation that each vehicle receives from the environment (the other vehicles in the surrounding area). The proposed approach is adaptive, which means that a novel collaborative Learning Automata (LA) makes decisions based on several parameters like density, mobility, and direction of motion that reflect the current state of the environment. The CTI is then determined for each process in the automaton. The automaton of each vehicle observes the activities carried out by other vehicles in its vicinity. However, the study assumes the completeness of information shared among neighboring vehicles. This does not hold for cITS systems as the vehicles communicate in a highly dynamic and harsh environment, which makes the communication between neighboring vehicles intermittent. Such sporadic communication causes a loss of context information. As such, the model could produce suboptimal accuracy when classifying the events as either legitimate or attacks.

To overcome the network instability and rapid topology change, [21] proposed a secured clustering algorithm that takes into account the vehicle's mobility during cluster formation.

Therefore, the clusters become more stable and each cluster head will be selected based on the vehicle's trust-level, which can easily be determined in such a stable environment. By employing the clustering approach, communication overhead is decreased because the broadcasting is reduced, which in turn decreases the data loss. The proposed model was tested on several attack types like selective forwarding, black hole, resource exhaustion, and Sybil attacks. Two levels of detection constitute the proposed solution: the Local IDS and Global IDS. Local IDS works on local nodes (vehicles) and monitors the neighboring vehicles. Global IDS, on the other hand, works on cluster head level and monitors the trustworthiness of cluster members. The decision is then taken globally at RSU level based on the Trust Level of each vehicle. However, the proposed model was built based on the assumption that the communication between vehicles within a certain cluster as well as across the clusters is stable, which might not be the case in the harsh environments similar to cITS networks. The highly dynamic and intermittent connectivity of vehicular networks was investigated by [22]. A game theory-based multi-layered IDS was proposed to detect attacks targeting vehicular networks. The solution relies on a set of pre-defined roles along with neural networks for classifying the traffic into either benign or malicious. The relationship between IDS and attacker was formulated as a non-cooperative game based on Nash Equilibrium. To guarantee that attack detection will work accurately on higher densities and to improve the robustness of the IDS even with a small fraction of data, a distributed clustering approach is adopted to group the vehicles in the network into different stable clusters. However, the clustering approach is not suitable for the ephemeral nature of vehicular networks as the lifetime of these clusters is very short. Such a clustering approach is also unsuitable for neural networks-based classification as these classifiers become outdated very quickly.

A Privacy-Preserving Machine Learning-based Collaborative IDS for cITSs was proposed by [23]. The model utilizes the knowledge-based database (i.e. the attacks pattern database) of each vehicle to improve the accuracy of the IDSs in the other vehicles. Moreover, the model in each vehicle employs the labelled data of other nodes to boost its own training data. To preserve data privacy, the model utilizes the dynamic differential privacy to capture the privacy notation in the collaborative IDS and use it to build the dual variable perturbation, which protects the privacy of the training data by perturbing the dual variable. However, the reliance on the labels acquired from the other nodes renders the entire IDS vulnerable to falsified labelling as the malicious vehicles might manipulate these labels, which adversely affects the validity of the knowledge-base of the other vehicles. In addition, sharing the entire knowledge-base among the vehicles in the network adds a massive overhead given the highly dynamic and ephemeral-nature of cITS networks. Such overhead might as well lead to loss of useful traffic data. Table 2.1 summarizes the existing IDS solutions for cITSs.

Table 2.1: Existing IDS studies for cITS.

Paper	Research problem	Solution	Limitation(s)
[19]	Maintaining the integrity and authenticity of data exchanged between vehicles is not suitable for cITS as they are resource demanding, hence not suitable for QoS and QoE.	• A Cloud-based IDS, which communicates with several cluster heads. • Cluster heads are connected to a group.	•Dividing vehicles into clusters is not suitable for highly dynamic and ephemeral networks like cITS. •The cluster head in cITS has a short living time within the cluster, which makes it unreliable as a mediator between the cluster's vehicles and service provider.
[20]	Most of the existing solutions are attack-specific and focus on limited types of attacks.	Integrates a Collaborative Trust Index (CTI) into a classification algorithm in order to cover many types of attacks.	Assumes the availability of all information about the neighboring vehicles which is not suitable in cITS because of: •The intermittent communication •Insufficient information gathered due to the ephemeral nature of cITS.
[21]	Existing IDS solutions overlook network instability and high dynamicity of the smart vehicular networks.	•A secured clustering algorithm that considers vehicle's mobility. •Two levels, local IDS (LIDS) and global IDS (GIDS) for cluster stability.	•It assumes a stable communication between within cluster and between clusters. •Not suitable for dynamic and harsh environments like cITS.
[22]	Lack of tradeoff between gathering sufficient information and preventing the overburdening of IDS's logging component with a high volume of unnecessary IDS traffic.	A multi-layered game theory-based neural network IDS with a distributed clustering: •Groups vehicles into different stable clusters, to improve the robustness of the IDS.	The clustering is not suitable because cluster's lifetime is very short and becomes outdated quickly.

Continued on next page

Table 2.1 – continued from previous page

Paper	Research problem	Solution	Limitation(s)
[23]	Due to privacy-preserving concern, existing IDS solutions ignore the sharing of detection knowledge among neighboring vehicles.	Share the knowledge-based of each vehicle with other vehicles while preserving the privacy of each vehicle.	•Labels received from other vehicles could be falsified. •Additional overhead.

2.2 Misbehaviour Detection Solutions in cITS

Unlike IDS, the misbehavior detection focuses on identifying the threats originating from the participation vehicles inside cITS. These threats come from several sources like hijacked, rogue, and/or faulty nodes. While hijacked and rogue vehicles disrupt cITSs operations intentionally, faulty nodes cause such disruption unintentionally. In addition, the faulty nodes might as well be the result of other types of threats like intrusion attacks. Misbehavior detection is further categorized into data-centric detection and node-centric detection [1]. The data-centric misbehavior detection observes the data and messages exchanged between the participating nodes and performs several checks to identify the false information and suspicious contents. The node-centric misbehavior detection monitors a vehicle within the cITS based on several aspects like the number of messages it sends in a certain time period and correctness of the message's format. The following subsections elaborate more about data-centric and node-centric misbehavior detection in cITS.

2.2.1 Data-Centric Misbehaviour Detection Solutions

As pointed out previously, data-centric misbehavior detection focuses on the data and messages exchanged among the neighboring vehicles in cITS. By utilizing the correlated packets from different sources, the newly received packet is vetted against several criteria like consistency and plausibility to determine its trustworthiness. Using the consistency

check, for instance, the average of the previous speed readings recorded for a vehicle can be used to judge the newly reported speed value. Readings from the same vehicle and from the neighboring vehicles are used to determine the consistency of the new information.

One of the main characteristics of consistency-based detection is its limited reliance on domain knowledge, which makes it easy to design and implement [1]. Plausibility, on the other hand, employs a predefined model to verify whether the received message is in line with the underlying model. Vehicle's speed, for instance, could be verified against the law of physics, which makes it impossible for a vehicle to travel at a speed of 1000 km/hour, which exceeds the upper limit of the known speed for a moving object. Several models have been proposed for data-centric misbehavior detection in smart vehicles. In the study conducted by [9], a framework for the certificate revocation process within Vehicular Ad-Hoc Networks (VANETs) was proposed. The framework relies on the trustworthiness evaluation of the participating vehicles to identify and exclude the misbehaving vehicles from the network. Such trustworthiness is updated according to several trust metric values calculated based on the data received within BSM packets. However, the trust metric relies on the assessment of the neighboring vehicles that are assumed honest. This might not be necessarily true as a misbehaving vehicle can send false reports about its neighboring vehicles to reduce its trustworthiness level. In such a case, the trust metrics of an honest vehicle can be decreased, which allows the attackers to manipulate the context of the traffic situation within that vicinity.

As traffic density has an influence on several events and behavioral aspects of the vehicles, it can be used as a security indicator to assess the plausibility and consistency of the traffic information sent/received by the nodes within the cITS. As such, it is important to secure the traffic density computation. To address this problem, a study conducted

by [10] measured the local traffic density of vehicles in cITS using two independent sensors. The traffic density information is used as a security parameter to address the illusion attacks challenge when an attacker employs a ghost (hijacked) vehicle to send false information (event messages) using valid credentials and showing valid location information. These measurements are then combined to evaluate a certain traffic situation and detect misbehaving vehicles. The study calculated local traffic density as a ratio between the number of neighboring vehicles and the total distance between these vehicles. However, the study assumes that at least one of the two independent sensors remains intact and the attacker has no access to it, which does not hold for sophisticated attacks that can infiltrate and manipulate all sensors.

To detect position falsification attacks, [11] proposed a machine learning-based model that observes BSM and determines whether they contain false data or not. Two classifiers, Logistic Regression and Support Vector Machines, were used to build the model. The Vehicular Reference Misbehavior Dataset is used to train the LR and SVM classifiers. After training the model offline, online testing was carried out by submitting the new messages into the model to decide whether they are falsified or not. However, the highly dynamic nature of the network topology in cITS and frequent vehicle disconnection may render it impractical to train a machine learning model.

CA-DC-MDS proposed by [2] is a multi-faceted context-aware misbehavior detection scheme for smart vehicles. The scheme utilizes the spatio-temporal correlation of the consistency between the cooperative awareness messages to protect these messages against internal attacks. Dynamic thresholds are used as context references that reflect the non-stationary nature of such networks. These spatial and temporal contextual thresholds were calculated using both Particle and Kalman filters, respectively. However, the

study assumes that the majority of the nodes (vehicles) are benign, which might not be realistic as attackers can exploit the compromised nodes to attack other nodes in a way similar to botnets. In their study, [3] proposed the Hybrid and Multifaceted Context-aware Misbehavior Detection model to address the limitation of existing context-aware misbehavior detection, which assumes stationary noise and ideal communication. This does not hold in the highly dynamic and harsh environments like cITS. The proposed model replaced the static plausibility and consistency thresholds with dynamic context references adaptable to the changes in the network topology. These context references are built online using several statistical techniques such as Kalman filter, Hampel filter, and Box and Whisker. However, the model was built on the premise that the majority of vehicles are honest. This does not hold in the case of botnet attacks, which exploit the compromised vehicles to create a chain of rogue nodes, resulting in a majority of compromised vehicles in the neighborhood.

In another study, [3] proposed an ensemble-based misbehavior detection model that replaces the static thresholds of the context of driving situations used by the extant research into dynamic thresholds that are determined online. As such, the proposed model is able to cope with the dynamic nature of the network. Kalman and Hampel filters were used to spontaneously adjust these thresholds. The model was trained using the data from the statistical classifiers, context parameters, consistency, plausibility, and behavioral features. However, like the authors of the previous studies I discussed, [3] assume the majority are honest, which does not hold for the case of massive attacks that used the botnet strategy to launch a chain of attacks. Table 3 summarizes the studies pertaining to MDS in cITSs.

2.2.2 Node-Centric Misbehaviour Detection Solutions

Node-centric misbehavior detection assesses the vehicle based on its behavior and trustworthiness. For the behavioral aspect, the number of messages sent by the vehicle and the validity of the format of these messages are observed. The trustworthiness, on the other hand, relies on a vehicle's reputation and the voting to determine whether the vehicle is misbehaving. Voting assumes the majority are honest. In the study carried out by [24], trustworthiness was employed to evaluate the vehicle and determine whether it was misbehaving. Such evaluation was carried out in the fog layer of cITS based on both intrinsic and extrinsic factors. Intrinsic factors rely on the information collected about the vehicle in question like the number of accidents, engine statistics, mileage, and velocity. Extrinsic factors rely on the information about the environment in the vicinity of the vehicle like a road map, trajectory, and proximity to other vehicles. Principal Component Analysis is utilized to analyze these factors and calculate the trustworthiness of the vehicle. However, relying on the intrinsic factors might not be suitable in the case of advanced attacks that manipulate the vehicle's own data. Those advanced attacks can also manipulate the context surrounding the vehicle by creating a collaborative illusion attack that falsifies driving situation information exchanged between the neighboring vehicles. The study conducted by [25] proposed a solution that incorporates the data trust model and vehicle trust model to discover the falsified data and evaluate the vehicle's trustworthiness. Support Vector Machines (SVM) and Dempster Shafer Theory (DST) are the main components of the model. SVM is used to evaluate the message contents, vehicle's attributes, and credibility based on its data propagation behavior. DST is then used to combine different multiple trust assessments about a certain vehicle; based on this, the final trust value is calculated. However, the model is built based on the premise

that the mobility information messages and vehicle's attributes are stationary. Such an assumption is not suitable for ephemeral environments like cITSs, in which the network is highly dynamic and the communication between vehicles is not necessarily reliable, which, in turn, invalidates the model within a short period.

Fuzzy Misbehavior Detection System was proposed by [5]to identify selective forwarding attackers who behave normally and only drop the messages coming from the neighboring vehicles. It employs fuzzy clustering to categorize normal and attacker nodes into different clusters. To build these clusters, Fuzzy C-Means clustering algorithm was utilized. Membership to either cluster is determined by a threshold which was empirically defined. However, relying on a static threshold to identify the membership degree is not suitable for cITS due to the dynamic nature of vehicles. The selective flow sampling and entropy method was used by [26] to detect the misbehaving vehicles in real-time. The purpose of the selective sampling was to extract the maximum amount of information from a small set of packets of a particular flow. This is helpful for real-time detection where the collected data are limited. The entropy was then used to calculate the change in the data collected previously. The study observed that with a small fraction of flows, a lot of information could be perceived and used for accurate misbehavior detection. However, entropy calculation might not be accurate in light of the number of flows available at the real-time deployment, which adversely affects the accuracy of the proposed model.

The major drawback of node-centric misbehavior detection is the reliance on the behavioral aspects of the nodes (vehicles) and the disregard for the semantical aspect of the data sent and/or received by these nodes. Moreover, the premise of majority honest that node-centric uses to calculate the trustworthiness of the nodes is not always true as some advanced attacks like those launched by botnets start by creating a majority

in favor of the compromised vehicles. In addition, evaluating such trustworthiness is challenging due to the ephemeral nature of the cITS, particularly at the initialization stage. Furthermore, the reputation mechanism used by node-centric misbehavior detection solutions is susceptible to sudden misbehaving or faulty vehicles [1].

Table 2.2: Existing research in MDS for cITS.

Paper	Research problem	Solution	Limitation(s)
[27]	Maintaining only the trustworthy vehicles and removing the misbehaving ones is challenging.	• A Certificate revocation framework is proposed based on trustworthiness evaluation using trust metric values calculated on the data received within BSM packets.	• Revocation of the certificate from honest vehicles. •Creating majority of malicious nodes that can be used to manipulate the context of traffic situation.
[10]	Using the environment information around host vehicle to validate the plausibility of the information sent by a particular vehicle is ineffective when the misbehaving vehicle provides valid location information.	•Two (independent) sensors were used to observe the environment surrounding the vehicle. If one sensor is attacked, the information of the other sensor stays intact.	• Assumes that at least one of the two independent sensors remains intact. •Attacker could hijack both sensors.
[11]	Guaranteeing the trustworthiness of the data in the presence of dishonest and misbehaving vehicles that share false information is hard.	•Machine learning was used to identify falsified BSMs.	•The dynamic network topology in cITS and intermittent communication makes it challenging to keep an up-to-date model.
[28]	Existing solutions assume that the context of driving situation is stationary, which contradicts the dynamic nature of cITSs.	•Dynamic thresholds were used as context references. Such a context reference can cope with non-stationary nature of the networks.	•The study assumes that the majority of the nodes (vehicles) are benign, •Compromised nodes can be used to compromise other nodes and create a majority dishonest.

Continued on next page

Table 2.2 – continued from previous page

Paper	Research problem	Solution	Limitation(s)
[3]	Existing context-aware misbehavior detection solutions assume stationary noise and ideal communication, which does not hold in the highly dynamic and harsh environments.	•Dynamic plausibility and consistency thresholds were built online to be adaptable to the changes in the network topology.	•Assumes majority honest, which does not hold in the case of botnet attacks that exploit the compromised vehicles to a majority of the compromised vehicles.
[6]	Existing research uses static thresholds for the context of driving situations.	•Multi-faceted dynamic context thresholds were proposed to adapt with the change in the driving situation.	•Assumes the majority are honest, which does not hold in case of massive attacks that used the botnet strategy to launch a chain of attacks.
[25]	Existing reputation-based misbehavior detection solutions assume that vehicles with high reputation are always trustworthy, which might not be accurate once they get compromised.	•Data trust model and vehicle trust model were incorporated to discover the falsified data and evaluate the vehicle's trustworthiness.	•Not suitable for highly dynamic network. •The communication between vehicles is unreliable.
[5]	Detecting misbehaving vehicles that behave normally and only carry out less suspicious actions like dropping some packets is challenging.	•Clustering approach was used to categorize nodes to normal malicious. •Selective forwarding attackers who behave normally are identified.	•Relies on a static threshold to identify the membership degree, which is not suitable for cITS due to the dynamic nature of nodes (vehicles).
[26]	Existing solutions overlook the fact that the data acquired about driving situation at real-time are not sufficient, which degrades the detection rate and increases the false alarms.	•A selective flow sampling was integrated with the entropy to extract the maximum amount of information from a small set of packets of a particular flow and calculate the change in the data collected previously.	•Entropy calculation might not be accurate due to limited number of flows available at the real-time deployment.

2.3 Comparing the IDS versus MDS cITSs Approaches

Table 2.1 summarizes some of the existing solutions using Intrusion Detection systems, while Table 2.2 compares multiple Misbehavior Detection solutions in cITSs. It can be seen that most of these solutions assume that the data shared between the vehicles are reliable. However, attackers could compromise such vehicles and exploit and manipulate the information they share with other nodes. The aforementioned studies try to find discriminative patterns to distinguish the attack traffic from the normal one. Although such an approach is effective in detecting resource exhaustion and spoofing attacks, it is unable to detect the misbehaving nodes that share false information with the nodes in its vicinity. This kind of misbehavior could be exploited by advanced attackers to manipulate the security thresholds and profiles. In addition, most of the studies assume that neither attack nor normal behaviors change, which does not hold as the IoT environment is dynamic. Therefore, such solutions suffer from concept drift, which invalidates the built defined profiles. Moreover, these solutions do not consider the obfuscation techniques and strategies that sophisticated attackers and malicious software use to deceive the detection. Based on the literature reviewed above, most of the existing IDSs and MDSs proposed for smart vehicles overlook the dynamic nature of the networks that these vehicles rely on to communicate with each other as well as the RSUs. These studies assume that the network is stationary, which is not realistic as the vehicles are always on the move and the lifetime of the connection is limited. Building security solutions on such assumption has a negative impact on the underlying thresholds and knowledge base that those models rely on for detection. Similarly, existing security solutions assume that the context of the driving situation is stationary, which is not true for cITSs as the context is stochastic due to the dynamic nature of such networks.

Communication reliability is another assumption that existing security solutions count on when building the detection models. However, the ephemeral nature of the smart vehicles and the harsh environment in which they operate makes it challenging to collect noise-free and complete attack patterns. Therefore, the collected data might not be representative enough for building accurate detection solutions. Likewise, acquiring sufficient information from within the network in real-time is challenging as the attackers escalate the attacks gradually to divert attention and avoid the suspicion. In such a case, the complete attack trace might not be available until the attack comes to the end. To the best of our knowledge, no study has considered this attack attribute, which renders these solutions vulnerable to this type of attack.

Given the limitations of IDSs and MDSs in cITSs, there is still a need for robust and accurate security solutions that take into account the fundamental characteristics of smart vehicular networks like the context dynamicity, communication unreliability, and data unavailability. In addition, proposed solutions need to deal with the sophisticated and massive attacks that try to create majority dishonest nodes and manipulate the driving context in the network. Similarly, such a solution needs to be aware of the false information that the compromised and rogue vehicles could share with neighboring vehicles. This could be achieved by employing a robust situational assessment that takes into consideration the change of credibility and reputation of the vehicles based on the observed behavior and pattern within a certain context. In the next two chapters, we built a detection model that takes into consideration the evasive nature of the sophisticated attacks. We also addressed the challenge of data insufficiency and evasive attacks that the attackers carry out on vehicles in the road section and deceive the detection mechanisms.

2.4 Conclusion

In this chapter, the existing literature related to intrusion detection solutions for IoT was reviewed. The review was done with a brief history on general overviews of attacks and security threats imposed against IoT. Then, it delved into data exchanged by on-road vehicles, especially those pertaining to traffic safety and efficiency. After that, the research into IDS for IoT was detailed. Related techniques for the online detection techniques were elaborated with the focus on the limitations of existing techniques. Those issues and limitations are addressed in chapters 4, 5, and 6. In the next chapter, the methodology that have been followed by this study is detailed.

Chapter 3

Methodology

As discussed above, the problem that this study has addressed is the inability of existing IDSs to identify the attacks launched internally against cITS nodes. Due to some unrealistic assumptions about the reliability of the nodes within the cITS ecosystem and the trustworthiness of the data shared by these nodes. Attackers can compromise a legitimate node and manipulate the data shared with the other nodes. Such manipulation can corrupt the normal behavior of all nodes in the vicinity. Consequently, the assumption that the data shared between these nodes are reliable does not hold in that case, which degrades the detection performance. Some studies have tried to address this problem by introspecting the context of driving situation and building Intrusion Detection Systems (IDSs) based on the contextual data collected at real-time. These solutions adapt to the changes in the situation such that, when the topology changes, the model triggers the re-training process to accommodate the new data and re-adjust the security parameters. However, data collected immediately after the topology changes are limited and might lack sufficient patterns that serve as evidence for the misbehaving attacks. The early data contain many missing and immature values, which worsens when data dimensionality is high. Imputing the missing values based on immature early data is challenging. Such data insufficiency also adversely affects the ability of a detection model to estimate features

significance accurately. Furthermore, these solutions are built based on the premise that the attack is stationary, which does not hold as attackers can change their behavior to deceive the IDS.

To achieve the objectives of this research, three phases are involved in the design and implementation of the model: Pre-processing and Feature extraction, Feature Selection, and model training/testing. The first phase corresponds to objective (1), in which the data undergo several preprocessing steps including the standardization and normalization. During the pre-processing, the imputation of missing values is carried out. The outcome of this phase is contextual data without missing values. The second phase corresponds to objective (2), in which feature selection process is conducted to select the important features set relevant to the data exchanged between the cITS nodes. The outcome of this phase is the features selection technique. The third phase corresponds to objective (3), in which a deep learning-based intrusion detection model is built by training Deep Belief Network algorithm with the features set extracted at phase two. The outcome of this phase is the IDS model for cITS systems. Figure 3.1 shows the general architecture of the proposed model. It consists of three main components: data preprocessing and features extraction, feature selection, and model training/testing.

3.1 Phase 1: Data Pre-processing and Feature Extraction

Figure 3.2 represents phase 1 of the architecture of Multivariate Fuzzy Clustering-based Data Imputation (MFC-DI) technique. This phase relates to the first objective of this study and has three main components. Those components consist of data acquisition, standardization, normalization, and imputation. In cITS ecosystem, data is collected from the environment using a set of sensors that collect data from the neighboring vehicles and

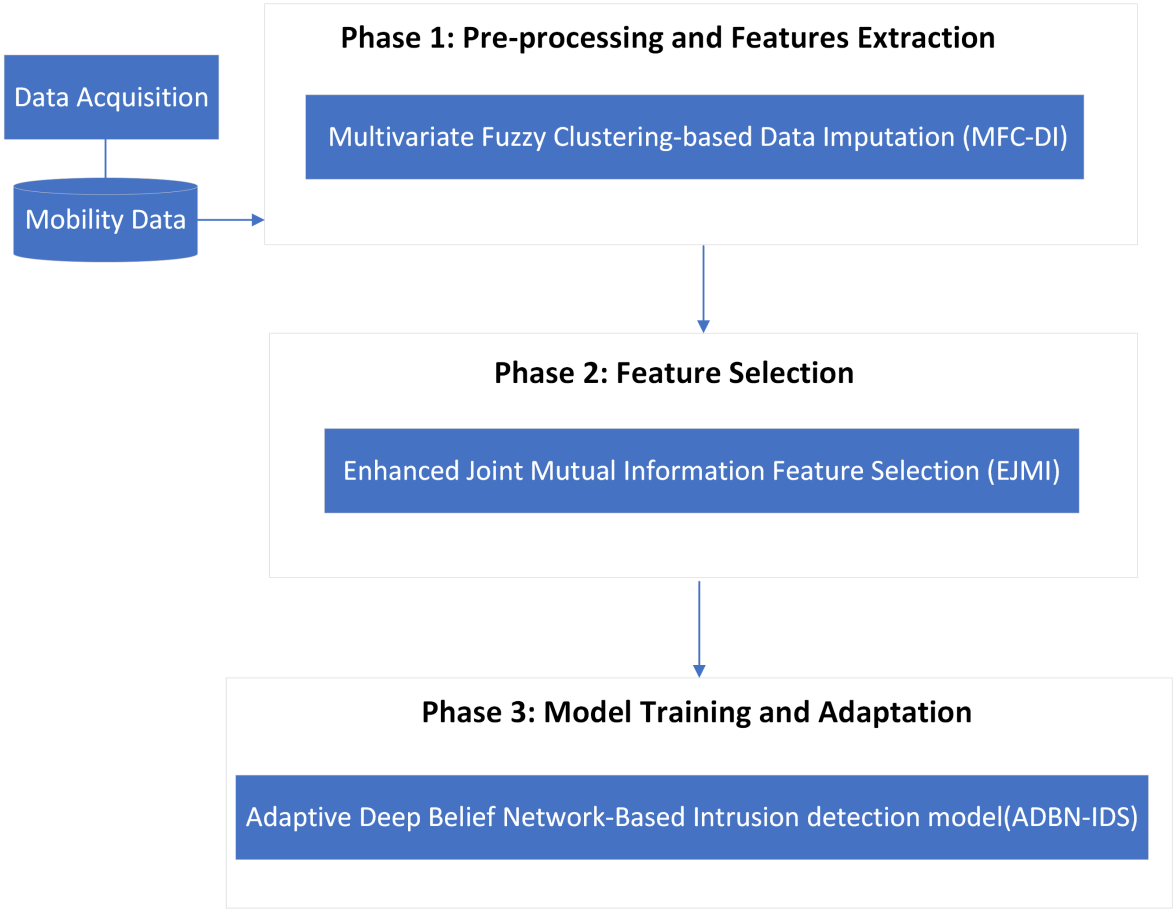

Figure 3.1: The general architecture of the proposed model.

store them either locally and/or in a central location. The MFC-DI is built based on the data collected. The raw data in the dataset consist of several components like contextual data, attack information data, and environmental-related data. The dataset is used as an impute for the imputation technique in order to estimate the missing values.

The purpose of data normalization is to put all values in the range between 0 and 1. This allows the MFC-DI to treat the attributes fairly and does not favor the attributes with higher ranges, like an acceleration ranging from 0 to 200 over other important attributes, or like an acceleration ranging from 0 to 15.84. After normalization, data standardization is carried out to transfer data so that means of zero and a standard deviation of 1 are achieved. This facilitates the modeling by creating a normal distribution for the data, which makes it easy for the algorithms to better understand the data. Furthermore, the purpose of the missing data imputation technique is to approximate the missing values acquired in the early phases of the new model's formation. Such estimation is carried out based on the values of the same attribute (univariate) as well as the values of other attributes (multivariate), especially those correlating strongly with that attribute. Unlike existing imputation methods which calculate the missing value based on the values of the same attribute, the proposed technique includes the values from the correlated attributes. A local-global fuzzy clustering technique calculates the missing values based on the data in the same attribute as well as the data from the correlated attributes.

To achieve MFC-DI, the correlation between attributes is determined by using Pearson correlation. Then, local-global clustering-based estimation is used to calculate the missing values. As the degree of correlation between an attribute and each one of the other attributes in the dataset can vary, a weight is given to its correlated attributes. This weight determines how much this correlated attribute contributes to the estimation of the

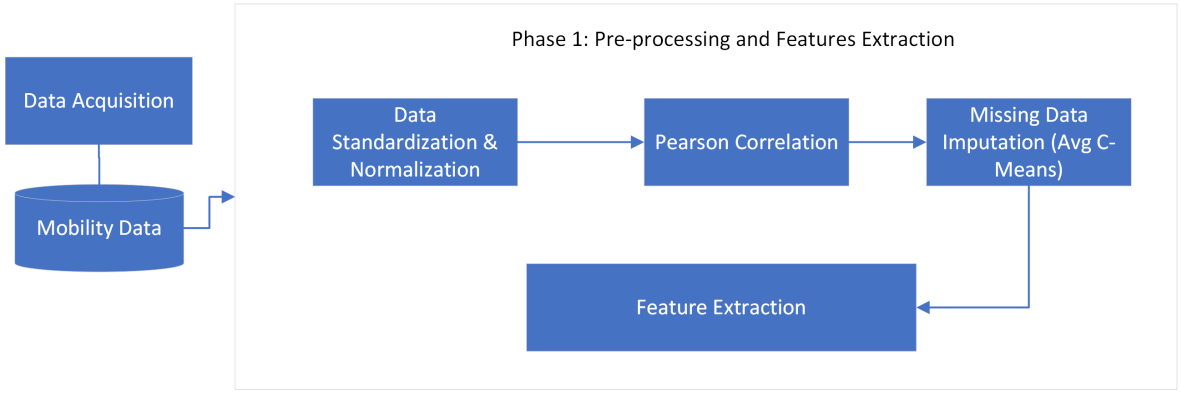

Figure 3.2: Data Pre-processing and Feature Extraction architecture.

missing value. By determining the correlation in the data distribution of two attributes, the missing value in one attribute is estimated based on the data distribution of the correlated attribute. This help to make an accurate calculation of the missing value using the immature values. In this context, strongly correlated attributes have more influence when calculating the missing value. That is, data distribution in the attribute in question follows the distribution of the value of its strongly correlated counterpart. As such, the data distribution of the counterpart attribute determines the data distribution of the attribute with missing values. After that, the features extraction is conducted to extract the semantical features based on the contextual data. The purpose of semantical data is to perceive the intrinsic characteristics of misbehaving nodes and overcome the polymorphic behavior of the attackers. The extraction of the semantics are introspected from the contextual information in the data. The outcome from the feature extraction step in this phase is the semantic features that represent the context during the MDS lifetime. In Chapter 4, you can find more information and a description of the Multivariate Fuzzy Clustering-based Data Imputation (MFC-DI) technique.

3.2 Phase 2: Feature Selection

During this phase, a feature selection is conducted to control the number of features extracted in the previous phase and avoid high data dimensionality while preserving the discriminative features. A Proportional Conditional Redundancy Coefficient (PCRC) is proposed to select the informative features from the dataset. PCRC is used in the Enhanced Joint Mutual Information Feature Selection (EJMI) for better feature significance estimation (see Figure 3.3). The goal function of EJMI consists of redundancy term and relevancy term. These terms play an important role in feature significance estimation as they control the trade-off between feature relevancy and redundancy. As such, it is imperative to calculate their values accurately. To do so, the PCRC is used to calculate optimal values of the redundancy coefficient in the goal function of the EJMI. By utilizing the PCRC for redundancy coefficient calculation, the EJMI becomes able to estimate the feature significance more accurately. This is because the PCRC has the ability to perceive the patterns from the early data more clearly and overcome the data insufficiency at the early stages of online adaptive models formation. In Chapter 5, you can find more information and a description of the Enhanced Joint Mutual Information Feature Selection (EJMI) technique.

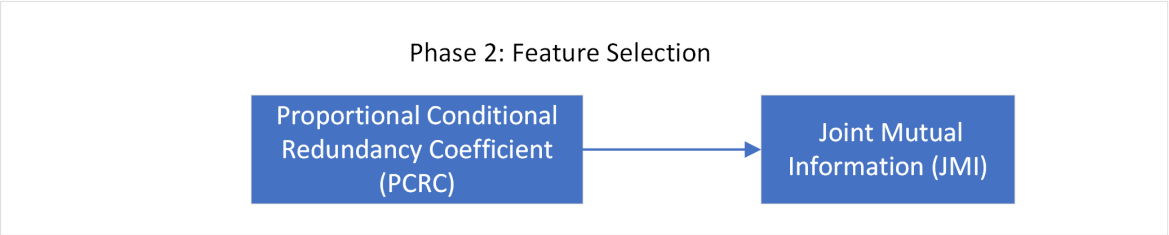

Figure 3.3: Feature Selection architecture.

3.3 Phase 3: Model Training/Testing

The third phase is related to the third objective of this study and involves two tasks: model training and model testing. While the former is carried out offline, the latter takes place online. That is, during model training, the features extracted and selected during the previous phase are used to train the Adaptive Deep Belief Network-Based Intrusion detection model(ADBN-IDS). The ADBN-IDS has been built using the Deep Belief Network (DBN) algorithm. These features are fed to the input layer of the deep neural network. A bi-variate moving average (BiMAV) method was developed and used in the (ADBN-IDS) to detect the (potential) diversion, in practice, from the existing threshold used by the detection model. Unlike existing methods that rely only on the values estimated at the output layer, the proposed technique uses the Bi-variate moving average method to correlate the change of output layer with averaged input variables. Such an approach provides precise change detection by avoiding the instantaneous changes that will eventually compromise the stability of the detection model. The proposed method prevents the unnecessary re-adjustment of security thresholds at the output layer of the DBN classifier thanks to the bi-variate-based moving average used to monitor and detect the change in the classification accuracy estimation. Figure 3.4 demonstrates how the ADBN-IDS model works. As mentioned before the model had been built using the Deep Belief Network (DBN) algorithm. After training the DBN model, it will be ready for prediction. the accuracy of the prediction is then examined by comparing the value of the BiMAV method (w) with the threshold (t). If the prediction exceeds the threshold, the model needs to be retrained. However, if the prediction is below the threshold, the current model is kept.

The ADBN-IDS is built using a training dataset that includes the semantic features

from phase 2. This enables the model to detect evasive attacks that change their behavior. Given the ability to detect evasive attacks, the ADBN-IDS is able to detect the novel (zero-day) attacks. Once tested, the ADBN-IDS can be used in a real-world deployment. During offline training, the ADBN-IDS uses the features of the DBN algorithm as derived from phase 2.. During online testing, detection accuracy is evaluated using a test set which the model has not seen before. When examining new instances, the same procedure of pre-processing and feature extraction are followed during the testing phase. The model then determines whether these instances are attacks. The outcome from this phase is the intrusion detection model. In Chapter 6, you can find more information and a description of the Adaptive Deep Belief Network-Based Intrusion detection model(ADBN-IDS). Table 3.1 summarizes the overall research approach including the research questions, objectives, methods, and evaluation metrics.

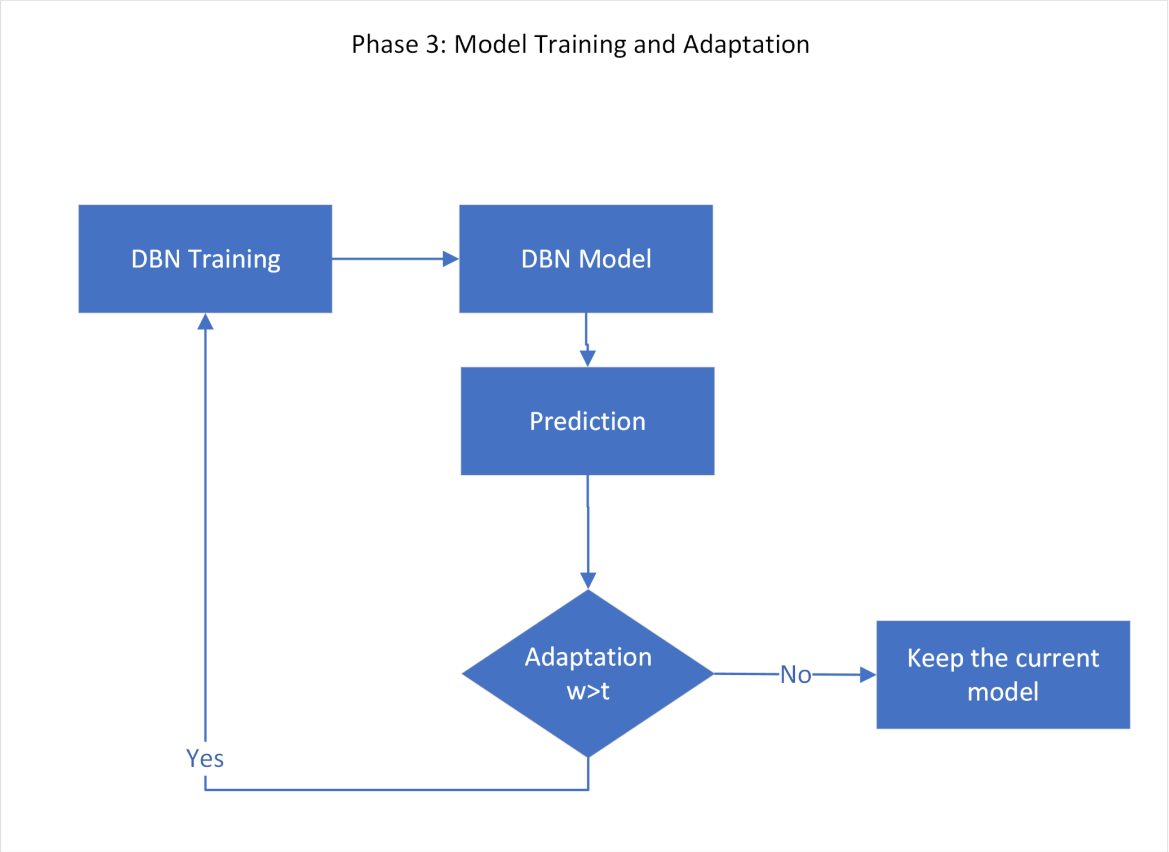

Figure 3.4: Model Training/Testing architecture.

Table 3.1: Research Plan.

Research Objectives	**Methodology**	**Questions Addressed**	**Solution Pursued**	**Performance Measures**
To propose a local-global clustering-based estimation technique to improve the accuracy of missing values' imputation in the mobility data collected at the early stages of model's formation.	The Pearson correlation technique will be employed to identify the correlated attributes and the degree of the correlation. Then, the averaged fuzzy c-means will be used to estimate the missing values based on values in the local attribute as well as the values on the correlated attributes.	How can statistical and clustering techniques be used to compensate the mobility data insufficiency and accurately estimate the missing and yet unobserved values during the early stages of model's formation?	Missing data will be imputed locally (univariate) based on the values of the same attribute (feature) and globally (multivariate) based on the values of the correlated attributes.	Accuracy (ACC), F1, False Positive Rate (FPR), and Detection Rate (DR).

Continued on next page

Table 3.1 – continued from previous page

Research Objectives	Methodology	Questions Addressed	Solution Pursued	Performance Measures
To propose an enhanced feature selection technique by incorporating a Proportional Conditional Redundancy Coefficient (PCRC) into the goal function of the joint mutual information feature selection to improve redundancy co-efficient value calculation, which consequently improves the feature significance estimation.	The PCRC was used to calculate the values of redundancy coefficient for the EJMI feature selection technique.	How can deep learning be used to accurately calculate the significance of features extracted from the incomplete mobility data during the early stages of model's formation?	Use the PCRC technique to calculate the values of redundancy coefficient in the goal function of the feature selection.	Model's accuracy.
To propose a deep learning-based Intrusion detection model by integrating the techniques proposed in (1) and (2) into a deep belief network structure.	A Bi-variate Moving Average (BiMAV) technique was integrated to the detection model so that the DBN model can adapt with the changes in the cITS system.	How can an adaptive approach be utilized to cope with the dynamicity and polymorphic nature of attacks?	A deep semantic-aware approach that observes the polymorphic patterns in the attack's behavior and re-adjust the security parameters accordingly.	Accuracy (ACC), F1, False Positive Rate (FPR), and Detection Rate (DR)

3.4 The Dataset

The dataset used in this research is the Next Generation Simulation (NGSIM) Vehicle Trajectories Dataset [29]. NGSIM is an open-source, publicly available dataset with a collection of real-world vehicles' trajectories collected by smart vehicles. The NGSIM dataset contains detailed vehicle trajectory data on southbound US 101 and Lankershim Boulevard in Los Angeles, CA, eastbound I-80 in Emeryville, CA, and Peachtree Street in Atlanta, GA [26]. Data in NGSIM were collected through a network of synchronized digital video cameras. NGVIDEO, a customized software application developed for the NGSIM program, transcribed the vehicle trajectory data from the video. This vehicle trajectory data provides the precise location of each vehicle within the study area every one-tenth of a second, resulting in detailed lane positions and locations relative to other vehicles. Moreover, NGSIM consists of many patterns representing different driving situations and driver behavior [29]. In addition, NGSIM provides high-quality contextual data that describe realistic real-world scenarios on different road sections [29, 30]. Particularly, NGSIM was built by collecting data from vehicles moving on a500-meter-long road section and on a seven-lane highway. For each vehicle, the data are collected (recorded) for 45 minutes using 16 sensors. Each record in the dataset contains a set of basic elements regarding the vehicle like position, speed, time, direction and acceleration.

The dataset represents the ground truth information and each vehicle represents an IoT node. In a real-world deployment, the dataset needs to be fed to each IoT node. That is, each node should have a copy of the dataset to run its own applications and adjust its communication or driving behaviour. As such, the collection of accurate and reliable context information is crucial. The context information in the dataset combines two types of messages: Cooperative Awareness Message (CAM) and Decentralized Environmental

Notification Message (DENM) into Basic Safety Message (BSM). While CAMs are sent periodically, DENMs are event-driven and are only sent when an event has occurred. The CAM consists of information about the vehicles like position, size, speed and steering wheel angle.

In contrast, DENM contains information about a certain event like lane changing and sudden braking. BSM will be used when discussing the combination of CAM and DENM messages. The first part of BSM, as well as CAM in the European standard, carries information about position, heading, speed, acceleration, steering wheel angle, vehicle role, vehicle size and status of vehicle light [1]. Unlike the first part of BSM that is included in all BSM messages, the second part of BSM (which corresponds to DENM in the European standard) is included only when an event happens.

3.5 Experimental Environment Setup

To implement the different components of the proposed model and evaluate its performance, the development and experimental evaluation are conducted using several tools and software including the Python, TensorFlow, Scikit Learn, SKFeature, and Numpy. These tools and libraries are all included in the Anaconda development platform. Meanwhile, the preparation of data samples, implementation of algorithms, and the analysis of the results are carried out on a machine with Intel(R) Core (TM) i7-4790 CPU @ 3.60 GHZ and 16 GB RAM. This relatively low powered working environment underscores the efficiency of the our model and approach.

3.6 Evaluation Metrics

To evaluate the performance of the proposed IDS for cITS, this study uses accuracy, precision, and F-measure as they are common metrics widely used by the extant research.

In addition, the approximation error of the proposed IDS model is measured by false positive rates and false negative rates. These measures are well known and commonly used in existing research reports as well. Equations 3.1, 3.2, 3.3, 3.4, and 3.5 are used to calculate the detection accuracy, detection rate, precision, false positive rate, and the F measure, respectively.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.1}$$

$$DR = \frac{TP}{TP + FN} \tag{3.2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3.3}$$

$$FPR = \frac{FP}{FP + TN} \tag{3.4}$$

$$F1 = \frac{2 \times \text{ Precision } \times \text{ Recall}}{\text{Precision } + \text{ Recall}} \tag{3.5}$$

where TP, TN, FP, FN denote the true positive, true negative, false positive, and false negative respectively.

3.7 Summary

In this chapter, the methodology used to achieve the research objectives is described. We start by revisiting the research problem and solution concept. The research plan that relates the research objectives with the methodology and solution is elaborated. The three phases of the proposed model development are discussed. Then, the dataset used in this research is introduced. The experimental environment is described. The tools and software used to carry out the implementation are also presented. The performance metrics used to evaluate the accuracy of the proposed model and compare the results

with other reported work. In the next three chapters, the design and implementation of the objectives are fully described.

Chapter 4

Deep Learning to Improve False Data Injection Attack Detection in Cooperative Intelligent Transportation Systems

This chapter is close to exactly as was published in October 2021 at the (IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)) [31]. This research addresses the first objective as described above, which is "To propose an local-global clustering-based estimation technique to improve the accuracy of missing values' imputation in the mobility data collected at the early stages of the model's formation." This technique can impute the missing values in an accurate manner when number of adjacent missing data is high, and this paper answered the following question:

Q.1 : How can statistical and clustering techniques be used to compensate the mobility data insufficiency and accurately estimate the missing and yet unobserved values during the early stages of the model's formation?

4.1 Abstract

This work proposes a local-global fuzzy-clustering feature extraction scheme for detecting False Data Injection Attacks (FDIA). In this scheme, the data undergo several pre-processing steps including missing values imputation based on the local and global fuzzy-clustering correlation approach. There are four main components of the proposed method: i) data acquisition, ii) standardization, iii) normalization, and iv) imputation (as described in Chapter 3 especially Figure 3.2). To evaluate the performance of this scheme, the NGSIM dataset (described below) was used. This dataset contains data acquired from the environment using a set of sensors that collect data from the neighbouring vehicles. The results show that the accuracy of models trained using said features extracted by the proposed scheme was higher than those proposed by the related studies. This indicates that the local-global fuzzy clustering data imputation approach proposed by this study can estimate the missing values better than existing techniques based on an exhaustive literature review.

4.2 Introduction

Vehicles in cITS are equipped with many sensors and actuators, by which data can be collected and shared with other neighbouring vehicles. These sensors include, but are not limited to, speedometers, acceleration, GPS, and Ultrasonic sensors. The data exchanged between the vehicles are contextual, event based, and situational [32, 33]. These data can be perceived from the contents of the messages exchanged between the drivers sharing the road segment [34, 35]. The vehicles exchange several types of data containing related driving environment characteristics, vehicle profile, and road conditions [36]. This includes speed, positions, acceleration, direction, driver status, and road status among many other

safety-related information. Such data can be utilized by many applications, especially those related to road safety, traffic efficiency, and fuel consumption [37]. Hence, the performance of these applications relies heavily on the quality of the data collected and exchanged among the nodes within the cITS [38–40]. However, data could be manipulated and falsified by attackers whose intention is to deceive the neighbouring vehicles to make wrong decisions like sudden brake or redirect all to a specific road section [3, 41]. Such attacks are carried out using malware that can evade security mechanisms and stay undetected [42, 43]. Consequently, many events of concern could be triggered such as accidents and congestion, which compromise road safety and traffic efficiency, and most certainly adversely affect the adoption of intelligent transportation systems [37].

Some studies have tried to address this problem by assessing the context of various driving situations and building Misbehaviour Detection Systems (MDSs) based on contextual data collected at drive-time. These solutions adapt to the changes in the situation such that, when the topology changes, the model triggers the re-training process to accommodate the new data and re-adjust the various security parameters. However, data collected immediately after the topology changes is limited and might lack sufficient patterns that serve as evidence for the misbehaving attacks. Early data therefore cannot be completely trusted because it likely contains too many missing and immature values, which worsens when data dimensionality is high. Consuming missing records/values based on immature early data is challenging. Such data insufficiency also adversely affects the ability of a detection model to accurately estimate a feature's significance.

The reliability, relevance, and trustworthiness of the data are the major quality factors that dictate the efficacy and feasibility of the cITSs [44]. However, assuring such data quality is challenging in an inherently highly dynamic and ephemeral cITS

environment [42, 45, 46]. This certainly follows from the implicit assumption that these studies make about the validity of the data collected at any point in the past. That is, the data used for building the predictive models are collected offline for model training. Therefore, the data is only relevant and valid for a short period of time before the context likely will change. After the context (where the data have been collected) changes, the predictive models become outdated. Despite some efforts to make these models adaptive [6, 9, 44], they were confined by data insufficiency during the time that proceed the context change [47].

Attack detection in the cITS ecosystem can be categorized into Entity-Centric detection (ECD) and Data-Centric detection (DCD) [1,48–52]. The focus of ECD is on identifying the vehicle(s) that behave suspiciously and/or broadcast falsified or manipulated data [53–57]. The ECD has been adopted for cITS, and a major part of it was replicated from legacy networks like MANETs [54, 58]. However, the high mobility, real-time and dynamic nature of the cITSs environment makes these solutions suboptimal [1]. ECD Solutions fall under two categories, behavioural-based or trust-based detection [49, 54, 59–61]. Vehicles using behavioural-based solutions are evaluated based on compliance with the rules and protocols [1]. For example, a misbehaving vehicle could broadcast messages at a rate higher than the normal range defined by the respective protocol [53, 62]. Therefore, the behavioural approach is unable to detect false data injection (FDI) attacks, especially those that are well-crafted that adapt to the dynamic nature of the cITS environment [46, 63]. Such kinds of misbehaviour and data manipulation is certainly difficult to distinguish from the real data.

The DCD, on the other hand, relies on a set of estimation criteria that examine the plausibility and consistency of the data exchanged between the neighbouring vehicles.

The data plausibility is estimated based on the degree of conformation to a predefined set of rules or model. These policies and/or models are derived from a set of well-known, agreed-upon rules like the law of physics that govern the maximum speed of vehicles on the ground to prevent exceeding a specific limit. Likewise, two physical objects cannot occupy (or overlap) the same space at the same time. A single object cannot be in multiple places at the same time [55, 64–67]. Such kind of behaviour, if observed, is not plausible and could indicate a suspicious activity or malfunctioning system. However, relying on such plausibility rules does not help in case of sophisticated attacks that obey these laws. Unfortunately, there are ways to bypass the plausibility-based countermeasure easily. Similarly, techniques that examine the consistency of the messages shared by neighbouring vehicles compare the received data with those that have been collected locally by the vehicle's sensors to check for corroboration. For instance, a vehicle sends data regarding its proximity to the traffic light that does not conform with the distance measured by the receiving vehicle. Therefore, this discrepancy is an indicator that the data received is not consistent with the local observations. Additionally, such consistency is prone to many types of errors and noise coming from the dynamic and stochastic nature of cITS's environment. As such, relying on consistency to identify the misbehaving nodes is limited to the ideal, noise-free and stable operational environment which may not be practical. Although some studies tried to incorporate adaptive modelling, such an approach is confined by the data insufficiency at the early stages of new situations that may not be tenable.

4.3 Proposed Methods

As discussed above, the problem that this study addresses is the inability of existing MDSs to identify the FDI attacks launched internally against cITS nodes due to some

unrealistic assumptions about the reliability of the nodes within the cITS ecosystem and the trustworthiness of the data shared by these nodes. Attackers could compromise a legitimate node and manipulate the data shared with the other nodes. Such manipulation could corrupt the normal behaviour of all nodes in the vicinity. Consequently, the assumption that the data shared between these nodes are reliable does not hold in that case, which degrades the detection performance. Some studies tried to address this problem by introspecting the context of driving situation and building MDSs based on the contextual data collected at real-time. These solutions adapt to the changes in the situation such that, when the topology changes, the model triggers the re-training process to accommodate the new data and re-adjust the security parameters. However, data collected immediately after the topology changes is limited and might lack sufficient patterns that serve as evidence for the misbehaving attacks. The early data contain many missing and immature values, which worsens when data dimensionality is high. Imputing the missing values based on immature early data is challenging. Such data insufficiency also adversely affects the ability of a detection model to estimate features significance accurately. Furthermore, these solutions are built based on the premise that the attack is stationary, which does not hold as attackers can change their behaviour to deceive the MDS.

To achieve this, we have developed a feature extraction scheme in which the data undergo several pre-processing steps, including standardization and normalization. After pre-processing, the missing values imputation is carried out. The scheme generates contextual data without missing values and is composed of four main components, data acquisition, standardization, normalization, and imputation. Unlike existing studies where the imputation takes place based on the data around the missing value(s) within

the attribute only, the technique proposed in this study also includes the values in the correlated attributes in such an estimation. Therefore, the proposed technique has the ability to deal with cases where the number of adjacent missing data is high.

4.3.1 Data Pre-processing

In the cITS ecosystem, data are collected from the environment using a set of sensors that capture data from the neighbouring vehicles and store them either locally and/or in a central location. The model is built based on the collected data. The raw data in the dataset consists of several components like contextual data, attack information data, and environmentally-related data. The dataset will be used as input for the imputation technique to estimate the missing values.

4.3.2 Data Normalization and Standardization

The data normalization, also called Min-Max scaling, is a scaling technique that puts all values in the range between 0 and 1. This allows the model to treat the attributes fairly and does not favour the attributes with higher ranges like speed (0 and 200) over other important attributes like acceleration, whose range falls between 0 and 15.84. Data normalization is conducted using the Min-Max formula as follows 4.1.

$$\bar{X} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{4.1}$$

where $X_{\max}$,$X_{\min}$ denote the maximum and minimum values of the attribute, respectively. On the one hand, if $X = X_{min}$, the numerator becomes 0, and $\bar{X}$ =0 the minimum. On the other hand, if $X = X_{max}$, the numerator will be equal to the denominator, and $\bar{X}$ =1. As such the value of the attributes are rescaled between 0 and 1.

After normalization, data standardization is carried out so that data points are centered around the mean with a unit standard deviation. As a result, the mean of zero and

the standard deviation of 1 are achieved. Data standardization is carried out using the following formula equation 4.2.

$$\bar{X} = \frac{X - \mu}{\sigma} \tag{4.2}$$

Where μ and σ denote the mean and standard deviation of the attribute's values. Both data normalization and standardization make the modelling easier by overcoming the challenge that early (i.e., immature) data brings when it does not follow the gaussian distribution. Therefore, it becomes easy for the algorithms to better understand the trends.

4.3.3 Missing values Imputation using Multivariate Fuzzy C - Means

As data are captured at the early stages after the online model's formation, these data consist of many missing values, which generates sparse data with many unknown and/or undefined values. Such missing data adversely affects the data quality and hence the models which are built based on such incomplete data sets. Therefore, missing data imputation is necessary. In this study, a Multivariate Fuzzy Clustering-based Data Imputation (MFC-DI) technique is proposed. The MFC-DI approximates the missing values in the data acquired during the early phases of our new model's formation. Such an estimation is carried out based on the values of the same attribute (i.e., univariate) together with the values of other attributes (i.e., multivariate), especially those which correlate strongly with that attribute. Unlike existing imputation methods which calculate the missing values based on the values of the same attribute, the proposed technique also includes the calculation values from the correlated attributes. Therefore, the MFC-DI estimates the missing values based on the data of the same attribute as well as the data

from the correlated attributes.

To achieve this, the correlation between attributes is first determined using bivariate Pearson Correlation (BC) method. The strength of the correlation is measured by the correlation coefficient, r, which is calculated using following equation 4.3.

$$r_{xy} = \frac{cov(x,y)}{\sqrt{var(x)}.\sqrt{var(y)}} \tag{4.3}$$

where cov(x,y) denotes the sample covariance of x and y attributes; var(x),var(y) denote sample variance of x and y, respectively. The expression cov(x,y) is calculated according to following equation 4.4.

$$cov(x,y) = \frac{\sum_1^n (X_i - \bar{X})(Y_i - \bar{Y})}{n-1} \tag{4.4}$$

where $\bar{X}$ and $\bar{Y}$ denote the mean for the attributes X and Y, respectively; n denotes the number of elements in both attributes. Moreover, the variance var(x) and var(y) are calculated according to following equation 4.5.

$$var(X) = \frac{\sum \left(X - \bar{X}\right)^2}{n-1} \tag{4.5}$$

4.3.4 Multivariate fuzzy c-means for missing data estimation

After data normalization and standardization, the multivariate fuzzy c-means estimation technique is used to approximate the missing data. As the degree of correlation between an attribute containing missing data and each one of the other attributes in the dataset can vary, a weight is given to its correlated attributes. This weight determines how much this correlated attribute contributes to the estimation of the missing value. Particularly, fuzzy c-means algorithm builds a set of clusters in which the likelihood that each data point falls within the boundary of a particular cluster has been estimated. The correlation co-efficient determined in the previous step is incorporated in the calculation of cluster

membership of each data point. That is, the membership of a datapoint was estimated based on the multiplication of the proximity of the datapoint from the cluster centroids multiplied by the correlation co-efficient between the data point and the nearest data points to the centroid.

Concretely, the proposed multi-variate fuzzy c-means imputation is designed based on the step-wise approach as follows. First, the number of clusters, n, is set, and the fuzziness index, m, is chosen between 1.25 and 2. This range of values have been chosen experimentally as they achieved the highest clustering precision. Then, the initialization matrix, U, is chosen as follows equation 4.6.

$$U = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{4.6}$$

After that the cluster centroids were calculated according to the following equation 4.7.

$$V_{ij} = \frac{(\sum_1^n (\gamma_{ik}^m * x_k))}{\sum_1^n \gamma_{ik}^m} \tag{4.7}$$

where V_ij denotes the j^{th} cluster centroid, γ_{ik}^m denotes the membership of j^{th} data point to the k^{th} centroid, and n denotes the number of data points. As pointed out above, the γ_{ik}^m is calculated by multiplying the proximity of the data point from the centroid by the correlation value between the data point and the centroid, as shown in the following equation 4.8.

$$\gamma_{ik}^m = \frac{r_{ik}}{\sum_{j=1}^c (\frac{d_{ik}}{d_{ji}})^{\frac{2}{m-1}}} \tag{4.8}$$

where r_{ik} is the correlation co-efficient between i and j, c denotes the number of centroids, and d_{ij} denotes the Euclidean distance between two data points i and k, which

is calculated by d_{ik} in the following equation 4.9.

$$d_{ik} = ||x_i - v_k||^2 \tag{4.9}$$

The fuzzy c-means algorithm tries to minimize the objective function, J, as follows 4.10.

$$J = min(\sum_{i=1}^{n}\sum_{k=1}^{c}(\gamma_{ik}^{m})d_{ik}) \tag{4.10}$$

The data point containing missing values is then compared with those that have similar averaged proximity from all centroids, and the missing value is imputed by the average of the corresponding values of those data points in the same attribute.

4.4 Results and Discussion

The results obtained from the proposed MFC-DI technique are presented here. The experiments were conducted using the NGSIM and the detailed vehicle trajectory data was collected from different locations using a network of surveillance cameras. The vehicle trajectory data were transcribed from the video stream by NGVIDEO software. The vehicle trajectory data provides the precise location of each vehicle within the study area every one-tenth of a second, resulting in detailed lane positions and locations relative to other vehicles [1]. The dataset contains two types of messages, Cooperative Awareness Message (CAM) and Decentralized Environmental Notification Message (DENM) into Basic Safety Message (BSM). While CAMs are sent periodically, DENMs are event-driven, which means they are only sent when an event has occurred. The CAM consists of information about the vehicles like position, size, speed and steering wheel angle. In contrast, DENM contains information about a certain event like lane changing and sudden

[1]The dataset was analysed and modeled in a workstation with 16 GB RAM, 2 TB hard disk, Windows 10 operating system. The software used for data analysis and modelling includes Python (Anaconda distribution), Pandas, Numpy, Sci-kit Learn, Tensor Flow, Keras, and skfeatures. Additional Fuzzy C-Means packages were individually installed using pip utility.

Table 4.1: Performance comparison of the LR, SVM, and CNN using the data processed by the MFC-DI.

#Metric / Classifier	LR	SVM	CNN
ACC	0.882075	0.884713	0.906034
F1	0.984255	0.985806	0.987415
FPR	0.168177	0.17672	0.152147
DR	0.872513	.877789	0.896802

braking. BSM will be used when discussing the combination of CAM and DENM messages. The first part of BSM, as well as CAM in the European standard, carries information about position, heading, speed, acceleration, steering wheel angle, vehicle role, vehicle size and status of vehicle light [1]. Unlike the first part of BSM that is included in all BSM messages, the second part of BSM (which corresponds to DENM in the European standard) is included only when an event happens, to carry information about such an event.

To evaluate the performance of the proposed MFC-DI, the dataset after imputation was used to train several machine learning models, namely the Support Vector Machine (SVM), Logistic Regression (LR), and Convolutional Neural Network (CNN). The dataset was split into training and testing using the k-fold cross validation method where k=10. The training set was used to build the models, whereas the testing set was used to evaluate the performance of these models. Several performance metrics were used to measure model performance, namely accuracy (ACC), F1 measure, detection rate (DR), and False Positive rate (FPR). Table 4.1 shows the results of the three machine learning models trained with the data obtained from the proposed MFC-DI technique.

The results in Table 4.1 show that the performance that the CNN model achieved was

higher than other techniques in terms of accuracy (ACC), F1 and Detection Rate (DR). More specifically, the accuracies of CNN, SVM, and LR were 0.906034, 0.884713, and 0.882075, respectively. Likewise, the F1 values were 0.987415, 0.985806, and 0.984255 for CNN, SVM, and LR, respectively. Furthermore, the false positive rates (FPR) generated from the CNN, SVM, and LR were 0.152147, 0.17672, and 0.168177, respectively. Additionally, the detection rates (DR) were 0.896802, 0.877789, and 0.872513, respectively.

In Figures 4.1, 4.2, 4.3, and 4.4 below, the comparison shows the performance of the proposed MFC-DI and the Pearson correlation-based missing data imputation [68]. The comparison was conducted in terms of accuracy, F1, false positive rate, and detection rate. Both MFC-DI and Pearson correlation-based imputation techniques were evaluated using three machine learning algorithms, namely SVM, LR, and CNN. As the comparison shows, the performance of the proposed MFC-DI outperforms the Pearson correlation-based imputation technique in all metrics, i.e., ACC, F1, FPR, and DR, for all classification models except the FPR of CNN where Pearson correlation was the lower. The comparison also illustrates that, in both MFC-DI and Pearson-based imputation, the CNN scored the highest performance values for the ACC, F1, and DR, while the performance of LR was the lowest.

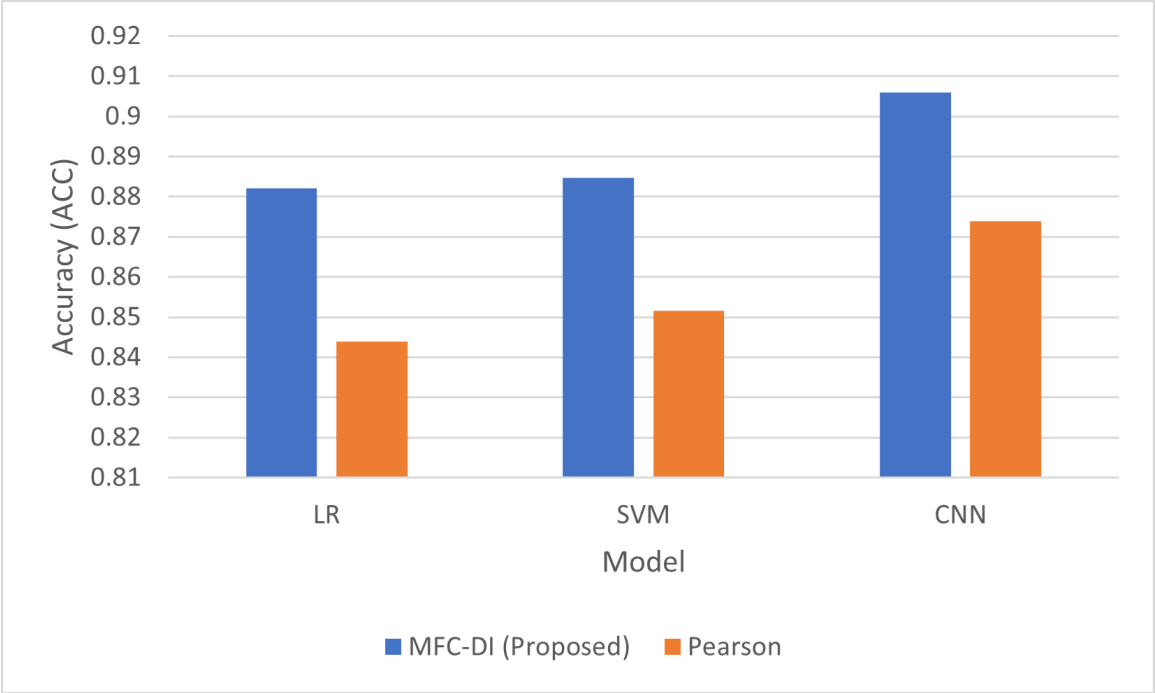

Figure 4.1: Accuracy comparison between MFC-DI and Pearson-Based imputation for the LR, SVM, and CNN.

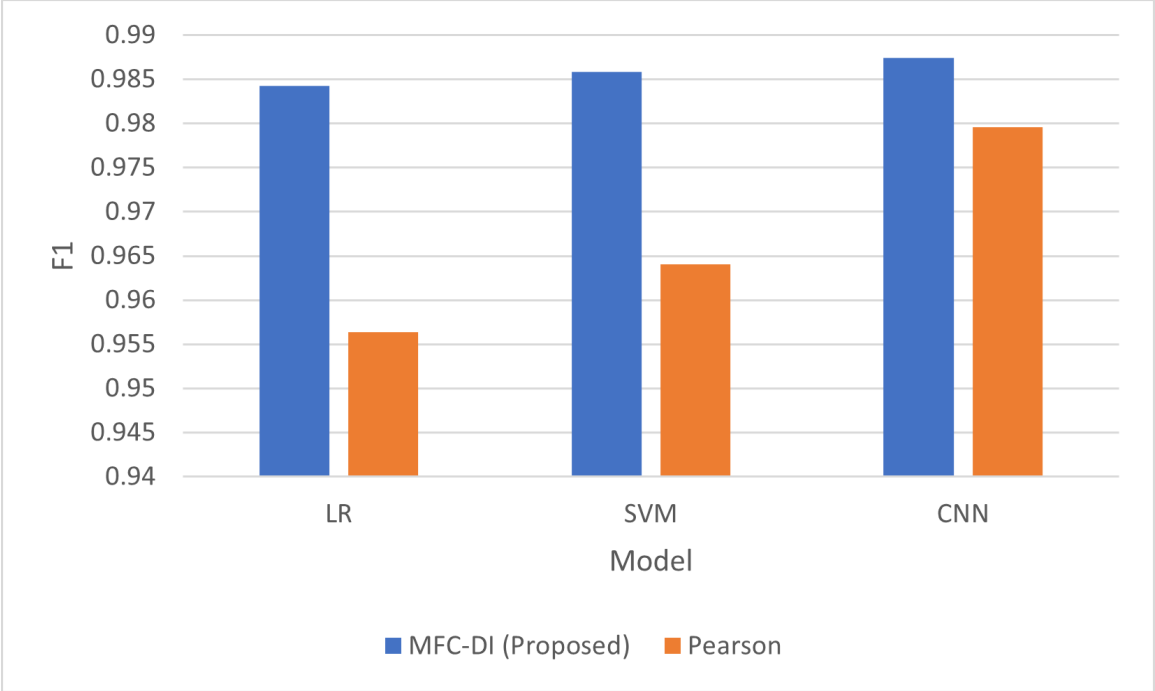

Figure 4.2: F1 measure comparison between MFC-DI and Pearson-Based imputation for the LR, SVM, and CNN.

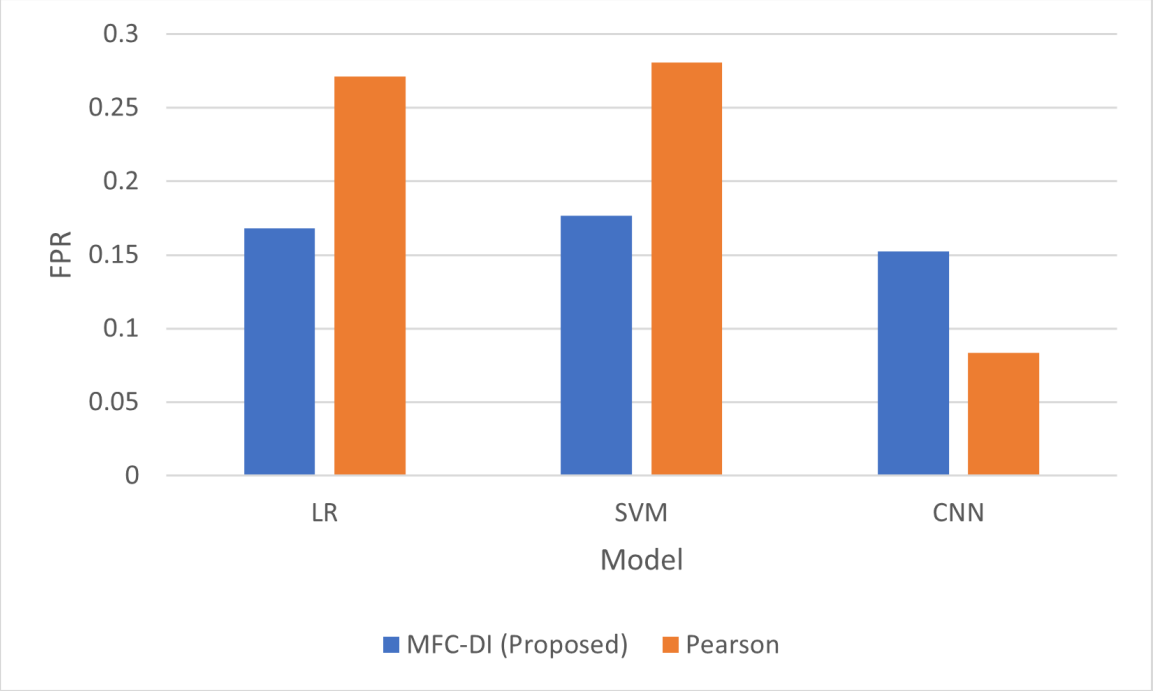

Figure 4.3: FPR comparison between MFC-DI and Pearson-Based imputation for the LR, SVM, and CNN.

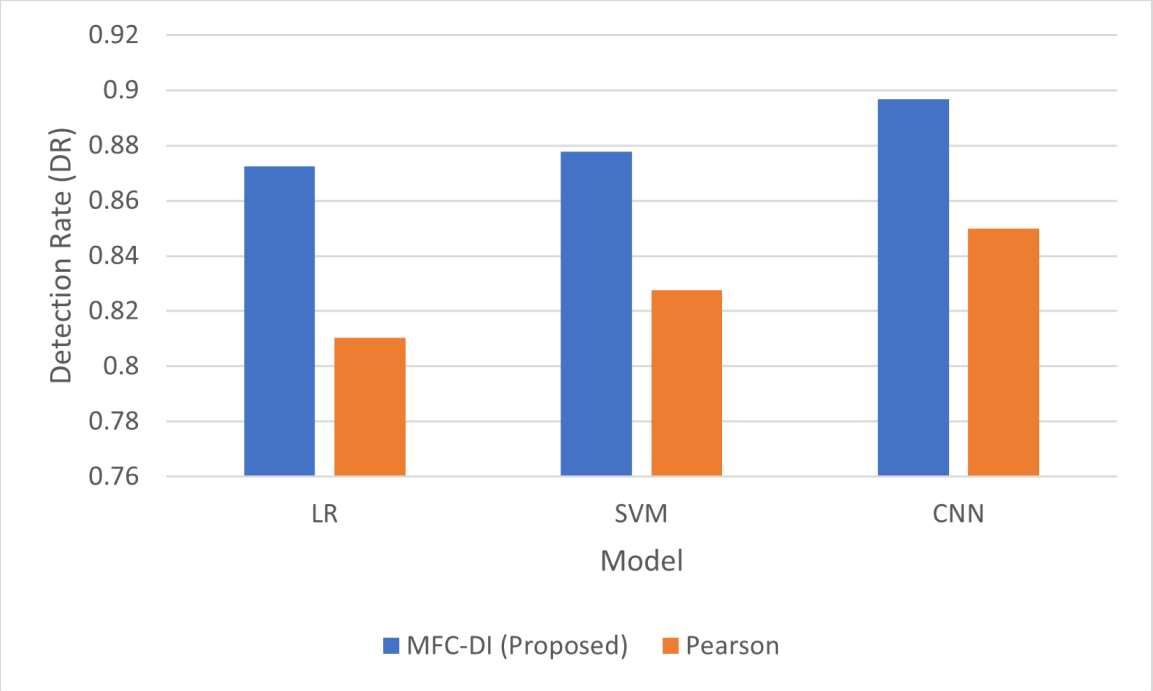

Figure 4.4: DR comparison between MFC-DI and Pearson-Based imputation for the LR, SVM, and CNN.

The results obtained from the proposed MFC-DI and the comparison with Pearson-based missing data imputation notably confirms better performance for almost all metrics which were evaluated except FPR for CNN. This is attributed to the multi-variate estimation used by MFC-DI, which involves not only the data around the missing value, but also the data with strong correlation and close clustering membership values similar to the data point containing the missing value. This indicates that the inclusion of the correlation coefficient in the calculation of membership value has improved the clusters' centroid-datapoint proximity estimation. This also helped with forming accurate and coherent clusters, which in turn improved the imputation of the missing data with more accurate values. The results indicate that the CNN model outperformed the other models from the literature. This is attributed to the ability of the hidden layers of the CNN network to perceive the hidden patterns in the data and to detect the weak relationships in the attributes. This indicates that the deep learning algorithms with multiple layers is robust to some inconsistencies that might exist in the data due to the potential for errors that might occur during the missing data estimation. It is worth noting that the FPR generated by MFC-DI was higher than the Pearson correlation when using CNN. The reason could be the effect of data insufficiency on data-hungry algorithms like CNN. This issue will be further investigated in our future study. From performance perspective, the MFC-DI inherits the complexity of Fuzzy C-Means algorithm in terms of computational complexity. This normally is expressed by O (NCT) expression, where N is number of instances, C is number of clusters, and T is number of iterations. As indicated by this big o notation, the time complexity is quite linear, which makes the proposed method practical for real-world implementation.

4.5 Conclusion

Herein a new model, the Multivariate Fuzzy Clustering-based Data Imputation (MFC-DI), is proposed. Our MFC-DI is a technique that approximates the missing values in the data acquired during the early phases of the new model's formation when the misbehaviour detection models in the cITS re-adjust to the environmental and contextual change. One caveat for adopting the multivariate approach is the computational overhead, especially for online models that adapt to the environmental and topological changes in the cITSs. This issue will be addressed in our future publication.

The technique incorporates the Pearson correlation into the estimation of clusters membership weights. The cluster membership was used to approximate the missing values in data points based on the counterpart values in the other data points that share the same proximity from the clusters. In such a way the missing data are determined by the multi-variate attributes rather than univariate attributes used by conventional imputation techniques. The results show that the proposed MFC-DI outperformed those that have been obtained in the past using conventional Pearson correlation. We expect this will help to improve the adaptive and online cITS models and increases the reliability of the data exchanged between the vehicles on the road section.

Chapter 5

Disrupting the Cooperative Nature of Intelligent Transportation Systems

This chapter is exactly as published in June 2022 at the (2022 IEEE World AI IoT Congress (AIIoT)) [69]. It addressed the first objective, which is "propose an enhanced feature selection technique by incorporating a Proportional Conditional Redundancy Coefficient (PCRC) into the goal function of the joint mutual information feature selection to improve redundancy co-efficient value calculation, which consequently improves the feature significance estimation." This technique helps to select relevant and non-redundant features during the early stages of model reconstruction, and this paper answered the following question:

Q.2: How can relevant features be selected from the incomplete mobility data during the early stages of model's formation?

5.1 Abstract

The emergence of Cooperative Intelligent Transportation Systems (cITS) simplifies the exchange of traffic situational information among vehicles within "close" proximity, which facilitates smooth traffic flow, reduces the congestion and saves energy. However, with such advantages come challenges represented by attackers who would compromise the

vehicle system components, spoof false telemetry and/or control signals causing serious problems such as congestion and/or accidents. There is need for security mechanism that can identify and detect such misbehavior in cITSs more dependably. Several studies have proposed Intrusion Detection Systems (IDS) for cITS depending on the contextual data exchanged between neighboring nodes. Those solutions rely on classifiers trained and readjusted online to reflect the dynamic nature of the cITS environment. These models are usually trained with a set of features selected based on insufficient data. This makes the feature significance estimation inaccurate due to data insufficiency collected from the online systems immediately after the model was updated. In this paper we address this issue by introducing a Proportional Conditional Redundancy Coefficient (PCRC) technique. The technique is used in the Enhanced Joint Mutual Information (EJMI) feature selection for better feature significance estimation. At each iteration, the PCRC increases the redundancy of the candidate feature proportional to the number of already-selected features while taking into consideration the class label. Such conditional redundancy is estimated for the individual features, which gives the feature selection technique the ability to perceive the attack characteristics regardless of the common characteristics of the attack. Unlike existing works, the proposed technique increases the weight of the redundancy term proportional to the size of the selected set. Consequently, the likelihood that a feature is redundant, given the class label, increases when more features are added to the selected set. By applying the proposed EJMI to select the features from the Next Generation Simulation (NGSIM) dataset of cITS, more accurate IDS has been trained as shown by the evaluation results. This helps to better protect the nodes in cITS against the cyberattacks (e.g., falsified data).

5.2 Introduction

By 2030, scientists anticipate that road accidents will be the major cause for deaths and injury [31, 53] within transportation systems. Vehicle accidents are also the primary cause for traffic congestion, which negatively impact the economy [70] and consumption. Billions of dollars are lost due to human error [71, 72]. Automation solutions introduced by the cooperative Intelligent Transportation Systems (cITS) have been proposed by researchers and industry to mitigate the problem. Such a solution can provide more contextual information that drivers (algorithms) can use when making decisions, which can improve road safety and traffic efficiency [73]. Given the cooperative nature of the cITS, individual vehicles can detect traffic anomalies in a nearly real-time manner based on the traffic information received from neighboring vehicles. These data exchanges unfortunately present an obvious security concern represented by misbehaving nodes (i.e. malicious or compromised vehicles) that share fake (dis-)information among participants within the cITS ecosystem [74].

The cITS, nodes are equipped with many sensors and communication devices that help collect and exchange information from the neighboring environment, including road and traffic conditions [75, 76]. The security is implemented in that the nodes analyze the data coming from neighboring node, based on which traffic anomalies can be detected [77, 78]. The cooperative nature of cITSs allows a wide range of applications including those used for safety, and traffic efficiency [79]. The accuracy of these applications depends on the reliability and thoroughness of the context information, which include position, velocity, and directions [80] [81] [82] [83] [84]. Nevertheless, due to the dynamic environment where cITSs operate, communication can be sometimes intermittent which leads to the potential for data falsification by malicious parties. Such falsification potential increases

the questionability of the trustworthiness of mobility information [85] [86] [87] [88]. This is due to cyber-attacks that compromise cITS nodes by causing them to share false information, which may lead to wrong decisions whose consequences could be disastrous to the traffic on the road section [85].

Security is therefore of major concern within the cITS ecosystem. Attackers can exploit the "cooperative nature" of applications in this environment to share false information. Wrong and/or inaccurate information can also be shared unintentionally, due to the malfunction of sensors on the cITS nodes or other intermittent communication errors. J. Grovert et al. [86] investigated various methods to create several attacks like position forgery, also known as illusion attacks. They found that with malicious interventions, legitimate traffic flow can be reduced by 80% while the legitimate packet delivery ratio can be reduced by 33%. Similarly, a study conducted by [87] revealed that illusion attacks degrade the routing efficiency of the cITS traffic. One explanation is that the misbehaving nodes evade the legitimate system routes, causing the protocols to make wrong decisions [1,88]. Securing cITS is challenging because vehicles work in a tumultuous environment where attackers can instruct the compromised node to send false information about road conditions. For instance, it is difficult to prevent attacks involving simulation of false environmental constraints to fool vehicle sensors [1]. Consequently, detecting such a misbehaving node is crucial for a secure cITS environment.

As misbehaving nodes pose significant threats to cITS [88, 89], various intrusion detection approaches have been proposed. The intrusion detection methods can be categorized into entity-centric and data-centric approaches [90–92]. The entity-centric approaches differentiate between compromised and benign vehicles. In contrast, the data-centric intrusion detection approaches try to identify false messages regardless, of

the sending node. Both approaches can be combined such that a data-centric approach is used for short-term, real-time applications and for privacy preserving purposes [91]. The entity-centric approach is then used for long-term detection when sufficient data has been collected. The performance of such a hybrid approach depends on the quality of the data as received from neighboring nodes as well as the classification accuracy [55, 93, 94]. However, the quality of the information acquired and shared between nodes cannot be ensured in the dynamic and tumultuous cITS environment [2, 7]. Therefore, the intrusion detection solutions need to be contextually aware of both data quality and trustworthiness to avoid the high rate of false alarms or low detection rates. Despite some efforts to create context-aware intrusion detection solutions as proposed in [3, 6, 33], these solutions rely on classifiers where the relevancy of features representing the attack were not confirmed. The machine learning-based techniques presented in [3, 6, 25, 95] were trained with features selected regardless of the incompleteness of the underlying attack data. That is, the previous models used data that was collected in a short time, which leads to low detection accuracy. Therefore, this paper focuses on improving detection performance by developing a more precise feature selection technique that can identify the attack's latent behaviour more accurately.

5.3 Related Work

The ongoing research in IDS for cITS relies on data extracted from the contents of the messages exchanged between communicating vehicles as well as contextual metadata that describes the operating environment. Such contextual data in many studies are static, which does not conform with the dynamic nature of cITS. In such a dynamic environment, the vehicle's context changes continuously. Therefore, the predefined security thresholds become obsolete more frequently, a major drawback that existing IDS solutions

for cITSs suffer. Some studies proposed solutions for this problem, like the context-aware data-centric misbehavior detection scheme (CA-DC-MDS) developed by [2]. This solution overcomes some of drawbacks discussed here. The static thresholds have been replaced by a dynamic threshold statistically calculated using a contextual model, which is constructed and updated online. The sequential analysis of temporal and spatial correlation was conducted using Kalman and Hampel filters to assess the consistency of mobility data exchanged between neighboring vehicles. The Kalman Filter tracks mobility data from the neighboring vehicles, while Hampel Filter assesses the consistency of this data. Based on the proximity from the threshold, the message containing the data is classified as either normal or suspicious. However, this scheme assumes the data collected at early phases, after the model updated its profile, are sufficient for consistency assessment. This is unrealistic, as the contextual data that describes the new situation is still immature.

Another study, [3] proposed a Hybrid and Multifaceted Context-aware Misbehavior Detection model (HCA-MDS), which combines four components: data-collection, context-representation, context- reference construction, and misbehavior detection. The model relies on data-centric features representing the cITS environment context. The contextual-based reference model was implemented using Kalman Filter and Hampel Filter as unsupervised non-parametric statistical methods. The consistency analysis was carried out using temporal and spatial correlation on mobility data. The purpose of such an analysis is to re-adjust the upper and lower boundaries of the reference model, which makes the model adaptive to the highly dynamic vehicular context. The maliciousness of the vehicles is assessed locally based on the consistency, plausibility, and reliability values of their mobility data. However, some drawbacks related to features are still not resolved, which lead to low detection rate as reported by the authors. This is due to the high

similarity, and sometime overlapping features of both normal and malicious behaviour, especially at the early phases after the adaptation takes place where no sufficient data the represent the new context has been collected yet. Therefore, there is a need for more accurate feature selection technique that evaluate the relevancy of features with insufficient patterns. Such a technique should be able to identify the difference between the benign and malicious vehicles behavior in some operational conditions with substantial uncertainty like maneuvering behaviour that looks like abnormal. Inability to distinguish this increases the false alarms.

Lacking to an accurate feature selection mechanism was investigated by [6], where an ensemble-based misbehavior detection system called (EHCA-MDS) was proposed. The study focused on how to distinguish between normal traffic and malicious behaviour patterns. They also studied the adaptability challenge when the context changes within the cITS. However, insufficient context representation during the time frame that precedes the formation of a new version of the model was not investigated, which is one of the unresolved issues in current ephemeral systems. This issue makes the existing feature selection technique unsuitable for this type of online system. Unfortunately, the lack of enough patterns needed for accurate decisions hinders performance.

Existing mutual information-based feature selection techniques adjust the redundancy coefficient inversely proportional to the size of the selected features set at each iteration [96, 97]. Such approaches decrease the belief in redundancy term each time a new feature is added into the selected set. Although this approach works well for data with full (complete) information observations about the attacks, it produces a sub-optimal feature set when dealing with data that lack enough observations [98, 99]. Reliance on comparison between the candidate feature and the common characteristics of all the already-selected

features in the selected set [97] is the weakness. Such common characteristics are difficult to perceive from incomplete data.

To this end, this paper proposes a Proportional Conditional Redundancy Coefficient (PCRC) technique by which, at each iteration; the redundancy of the candidate feature increases proportional to the number of already-selected features given the class label. Such conditional redundancy is estimated for the individual features, which gives the feature selection technique the ability to perceive the attack characteristics regardless of the common characteristics of the attack. Unlike existing works, the proposed technique increases the weight of the redundancy term proportional to the size of the selected set. Consequently, the likelihood that a feature is redundant, given the class label, increases when more features are added to the selected set. The intuition is that by individually comparing the candidate feature with ones already selected with respect to class label, the likelihood that the candidate feature is redundant to one or more of those features increases proportional to the size of selected set. The following sections elaborate on the design and implementation details of the PCRC as well as incorporating such with a mutual information feature selection technique.

5.4 Methodology

The mutual information (MI) is defined as the amount of information that two variables share about each other [97]. This is calculated according to equation 5.1 as follows.

$$MI(X;Y) = \sum_{y \in Y} \sum_{x \in X} d(x,y) \log \frac{d(x,y)}{d(x)d(y)} \tag{5.1}$$

where $MI(X;Y)$ is the mutual information between the vectors X and Y, d(x) and d(y) are the marginal distribution of x and y variables; and d(x,y) is the joint distribution. The equation 5.2 is the general representation that describes the calculation of the MI.

$$J(X_k) = MI(X_k;Y) - \beta \sum_{X_j \in S} MI(X_j;X_k) + \gamma \sum_{X_j \in S} MI(X_j;X_k|Y) \tag{5.2}$$

where $MI\left(X_k;Y\right)$ is the mutual information between the candidate feature X_k and the class label Y;$MI(X_j;X_k|Y)$ represents the conditional mutual information between the candidate feature X_k and the feature X_j in the selected set S given the class label Y; β and γ are the redundancy and conditional redundancy coefficients, respectively. The values of these coefficients range between 0 and 1. The equation 5.2 consists of two expressions, relevancy term 5.3 and redundancy term 5.4. Furthermore, the redundancy term consists of two sub-terms, namely the marginal redundancy represented by the expression 5.5 and the conditional redundancy represented by the expression 5.6.

$$MI\left(X_k;Y\right) \tag{5.3}$$

$$\beta \sum_{X_j \in S} MI\left(X_j;X_k\right) + \gamma \sum_{X_j \in S} MI\left(X_j;X_k \mid Y\right) \tag{5.4}$$

$$\beta \sum_{X_j \in S} MI\left(X_j;X_k\right) \tag{5.5}$$

$$\gamma \sum_{X_j \in S} MI\left(X_j;X_k \mid Y\right) \tag{5.6}$$

The mutual information makes trade-off between the relevancy and redundancy terms. This is achieved by fine-tuning at least one coefficients, i.e. redundancy β ,or marginal

redundancy γ. It turned out that the calculation of relevancy term always is same, which relies on calculating the relevancy between the candidate feature X_k and the class label Y. Therefore, the redundancy calculation is what makes difference in terms of features accuracy.

As can be inferred from expression 5.2, the values of the coefficients β and γ are essential for the relevancy-redundancy trade-off. The value of γ involves the class label into this calculation. This makes it more influential for feature significance estimation as the feature could be non-redundant with respect to one label and redundant with respect to another label. Existing feature selection techniques do not consider this and calculate the feature relevance regardless of the class label. Concretely, the small value of γ decreases the effect the marginal redundancy and; consequently; increases the feature's significance. Therefore, the PCRC is incorporated into the JMI. The proposed EJMI (Enhanced Joint Mutual Information) technique adopts the approach used in the mRMR and JMI techniques and calculates the mutual information according to equation 5.2. Unlike the traditional JMI that decreases the conditional redundancy weight inversely proportional to the number of features in the selected set (see expression 5.7). The proposed PCRC-based EJMI is proportional to the size of the selected set (see expression 5.8).

Concretely, PCRC calculates the redundancy weight using expression 5.9, where starts with low values and increases when the size of the selected set (S) increases. The intuition is that with more features added to the selected set, the likelihood a new feature for a specific label is redundant increases as well. Therefore, the redundancy weight must be increased. The proposed PCRC achieves this by putting the size of the selected set at the numerator so that the increase of the size will correspondingly increase the value of the

conditional redundancy coefficient, and consequently give more wight to the conditional redundancy term of the EJMI. The value in the denominator is fixed throughout the selection process and equals to the size of original set F.

$$\gamma = \frac{1}{|S|} \tag{5.7}$$

$$\gamma = \frac{|S|}{|F|} \tag{5.8}$$

where $|S|and|F|$ represent the number of features in the selected and original set respectively. Therefore, EJMI selects the informative features according to equation 5.9.

$$\begin{aligned} J\left(X_k\right) = MI\left(X_k;Y\right) - \frac{1}{|S|}\sum_{X_j \in S} MI\left(X_j;X_k\right) + \\ \frac{|S|}{|F|}\sum_{X_j \in S} MI\left(X_j;X_k \mid Y\right) \end{aligned} \tag{5.9}$$

The pseudo code of EJMI is shown in Figure 5.1. It calculates the significance of the features based on the equation 5.9, which considers both the original features vector F and S. EJMI selected the informative features according to equation 5.10 . As shown in the pseudo-code , $F = \{f_1, f_2, f_3, \ldots f_{n-1}, f_n\}$ represents a vector containing n number of features; V is temporary set that accommodates the features whose MI values have already been calculated; $S = \{s_1, s_2, \ldots, s_t\}$ represents the selected set with t number of features. EJMI started by creating the empty sets V and S. Then it calculates the MI value for each feature f_i in F. After that, the MI values were used to rank the features. Then, the features are stored in the set V in ranked manner. After that, the feature, v_k, in V with max(V,MI) was simultaneously removed from V and added into S. The next

Figure 5.1: Pseudo code of our EJMI feature selection technique

Pseudo Code 1: EJMI Technique

Input: $F = \{f_1, f_2, \ldots, f_n\}$ original features vector; C class label,number of features required (t).

Output: $S = \{s_1, s_2, \ldots, s_\rho\}$ the selected set.

1: $V \leftarrow \emptyset; S \leftarrow \emptyset$
2: for each feature $f_i \in F$:
3: $\quad v_i = MI\left(f_i; C\right)$
4: $\quad V \leftarrow V \cup v_i$
5: $v_k \leftarrow \max(V, MI)$
6: $S \leftarrow v_k; V \leftarrow V \backslash \{v_k\}$
7: for $\forall \left(v_j, s_m\right)$ with $v_j \in V$ and $s_m \in S$
8: $\quad$ compute $MI\left(C; s_m \mid v_j\right)$
9: $\quad s_p = \underset{v_j \in V}{argmax} \left[\sum_{s_m \in S} \text{EJMI}\left(C; v_j \mid s_m\right)\right]$:
10: $\quad V \leftarrow V \backslash \{s_p\}$
11: $S \leftarrow S \cup \{s_p\}$
12: Repeat $8 - 11$ while length $(S) \leq \tau$

feature v_p was chosen according to equation 5.10.

$$J\left(X_k\right) = MI\left(X_k; Y\right) - \frac{1}{|S|} v_p = \underset{v_j \in V}{argmax} \left[MI\left(v_j; C\right) - \frac{1}{|S|} \sum_{s_j \in S} I\left(v_k; s_j\right) + \frac{|S|}{|F|} \sum_{X_j \in S} MI\left(X_j; X_k \mid Y\right)\right] \tag{5.10}$$

At each iteration, the feature v_j from V that produces the highest MI value given the class label with respect to the already-selected features was added to the selected set. When the number of features in the selected set reaches the threshold t, the selection process is stopped.

5.5 Results and Discussion

In this section, the efficacy of the proposed PCRC technique for Joint Mutual Information Feature Selection is evaluated. We begin by detailing the dataset and then the environment with which the experiments have been conducted. The results obtained after applying the proposed techniques are presented along with a comparison to related work.

5.5.1 Dataset

In this section we describe the dataset used to evaluate the performance of PCRC. Herein, we used the same evaluation procedures employed in [72,73]. This includes data collection and preprocessing, noise injection, message loss and malicious nodes simulation. The dataset used for this phase is the Next Generation Simulation (NGSIM) [74] (see Chapter 3 for a discussion of the phases). NGSIM contains around 5000 node trajectories. The trajectories and vehicles were also simulated to capture the context data and share them with neighboring vehicles. The simulation has been done in Python. The NGSIM [74], was chosen as it is commonly used in similar studies to validate IDS models. The dataset contains roughly 5000 vehicles with many traffic scenarios that describe driver behavior, vehicle density, velocity, and traffic flows.

Noise was injected to simulate the harsh and tumultuous cITS environments. This noise included static noise, dynamic noise, and dynamic correlated noise. The static white noise comes in open sky environments where there is no signal deterrence like those in rural areas or a desert where the vehicle has access to GPS. The dynamic noise occurs in cloudy environments. Correlated noise occurs in downtown environments where tall buildings and bridges prevent the signal from reaching neighboring vehicles [2, 3, 6]. On the one hand, the static and dynamic white noise are normally distributed. Moreover, the static noise has fixed variance while dynamic noise varies based on time. The correlated noise is modeled based on the random walk approach [2, 3, 6, 99].

Communication loss is a substantial challenge due to the mobile nature of cITS nodes [92, 100, 101]. The rate of such loss depends on several factors such as traffic density, vehicle velocity, and road obstacles. As vehicles speed up and slow down, they meander in and out of communication range. Such behaviour causes signal/data loss.

Moreover, the intermittent connections increase inversely proportional to traffic density. Thus, when traffic density increases, congestion happens while at the same time, highly dynamic context transmissions in safety applications occurs every 100ms. These leads to channel congestion and thus data loss. However, data reliability is the main issue that adversely affects the accuracy of the IDSs [70]. One fix to this is the adoption of local context prediction where each vehicle estimates its own context data regularly. Based on a prediction error, the decision to send the current contextual data is executed. This helps to reduce the congestion which in turns reduces data loss.

Given these constraints, malicious nodes that send false context data to negatively impact road safety and traffic efficiency were simulated as suggested by [80, 81]. The attacks include sudden or random position jumping, Sybil attack, inaccurate movement patterns, and consistency attacks. The most challenging attack is the consistency attack in which the attacker tries to generate consistent but fake vehicle trajectories to cause traffic deception (i.e., an illusion) that degrade applications and network performance. These attacks were implemented, based on works proposed by [3, 6, 33, 80, 81].

5.5.2 Experimental Results and Analysis

Tables 5.1, 5.2, and 5.3 show the accuracy of the features selected by EJMI (using the proposed PCRC technique) and the comparison with those selected by existing techniques (i.e., mRMR, MIFS and JIM). Several classifiers have been used to test the accuracy of the selected features, namely the Deep Neural Network (DNN), Support Vector Machine (SVM), and Logistic Regression (LR). The performance was evaluated in terms of accuracy, which is calculated based of the well known equation 5.11.

$$accuracy = (tp + tn)/(tp + tn + fp + fn) \tag{5.11}$$

where tp, tn, fp, fn are true positive, true negative, false positive and false negative,

respectively [63]. The 1st column in each table lists the accuracy of the proposed EJMI, while the 2nd, 3rd, and 4th columns list the accuracy of the related feature selection technique, i.e. MIFS, mRMR, and JMI. The tables' columns are used to list feature sets with different sizes. The feature sizes range between 5 and 23 incremented by 3. The results show that the proposed EJMI achieved higher accuracy when used to train the three classifiers (ACC - 2.6% than DNN, 2.0% than SVM, 3.1% than LR). This is attributed to the ability of PCRC (incorporated to EJMI) to calculate the conditional redundancy term more precisely when insufficient data are presented right after the model adapts to the changing environment. This in turns improves the ability of EJMI to estimate the accuracy of the features significance more precisely. By training the classifiers with relevant features only, the detection model becomes more accurate and the likelihood that it over fits due to highly dimensional data will be decreased. Such dimensionality reduction also improves the efficiency of the detection model and makes it suitable for online application.

During the experimental evaluation, several thresholds (5, 8, 11, 14, 17, 20, 23, and 25) have been tested to pinpoint the suitable number of features that result in high detection accuracy. The results show that accuracy increased when more features were added, until the number of features reached 14. After that, the increase of the accuracy became less gradual. This is applied to the three classifiers. The reason being, that the model needs sufficient features to make correct decisions. But when the number of features exceeds a certain limit, the model would suffer from high variance that makes it prone to over fitting. The situation exacerbates when the coming observations lack the sufficient attack patterns necessary for clear and accurate decisions. This would result in a model that can only recognize the patterns that it has seen before, and if new patterns that have less

similarity with the known ones is encountered, the likelihood that the model could miss the true classification becomes high.

Table 5.1: Accuracy comparison between the proposed PCRC incorporated into the EJMI for feature selection and the existing feature selection techniques using DNN

Classifier / Features	EJMI	MIFS	mRMR	JMI
5	0.895	0.87	0.871	0.891
8	0.907	0.87	0.867	0.883
11	0.931	0.89	0.899	0.905
14	0.959	0.904	0.915	0.923
17	0.961	0.914	0.929	0.939
20	0.974	0.909	0.936	0.947
23	0.97	0.904	0.931	0.943
25	0.967	0.901	0.914	0.94

Figures 5.2, 5.3, and 5.4 show the comparison between the proposed EJMI and related feature selection techniques. It can be observed that EJMI achieved high accuracy when the features were used to train the DNN. The accuracy started from 0.895 when trained by 5 features and increased to 0.974 when trained by 20 features. Note that when the number of features increased to 23, the accuracy dropped. The same observation is true with the SVM model whose accuracy dropped from 0.949 to 0.942 and LR whose accuracy dropped from 0.952 to 0.951 when the number of features increased to 23. The reason being that over fitting occurs when the number of features increase. This supports our argument, pointed out above, in that the increase in the number of features increases the data dimensionality, and consequently the model becomes prone to over fitting.

Table 5.2: Accuracy comparison between the proposed PCRC incorporated into the EJMI for feature selection and the existing feature selection techniques using SVM

Classifier / Features	EJMI	MIFS	mRMR	JMI
5	0.879	0.852	0.856	0.871
8	0.889	0.854	0.852	0.87
11	0.913	0.872	0.884	0.882
14	0.941	0.885	0.9	0.91
17	0.943	0.895	0.914	0.927
20	0.949	0.882	0.919	0.934
23	0.942	0.88	0.915	0.93
25	0.938	0.88	0.911	0.926

Table 5.3: Accuracy comparison between the proposed PCRC incorporated into the EJMI for feature selection and the existing feature selection techniques using LR

Classifier / Features	EJMI	MIFS	mRMR	JMI
5	0.881	0.855	0.852	0.875
8	0.893	0.854	0.848	0.866
11	0.917	0.865	0.88	0.888
14	0.944	0.888	0.881	0.906
17	0.947	0.899	0.91	0.923
20	0.952	0.905	0.913	0.92
23	0.951	0.905	0.911	0.917
25	0.945	0.904	0.906	0.912

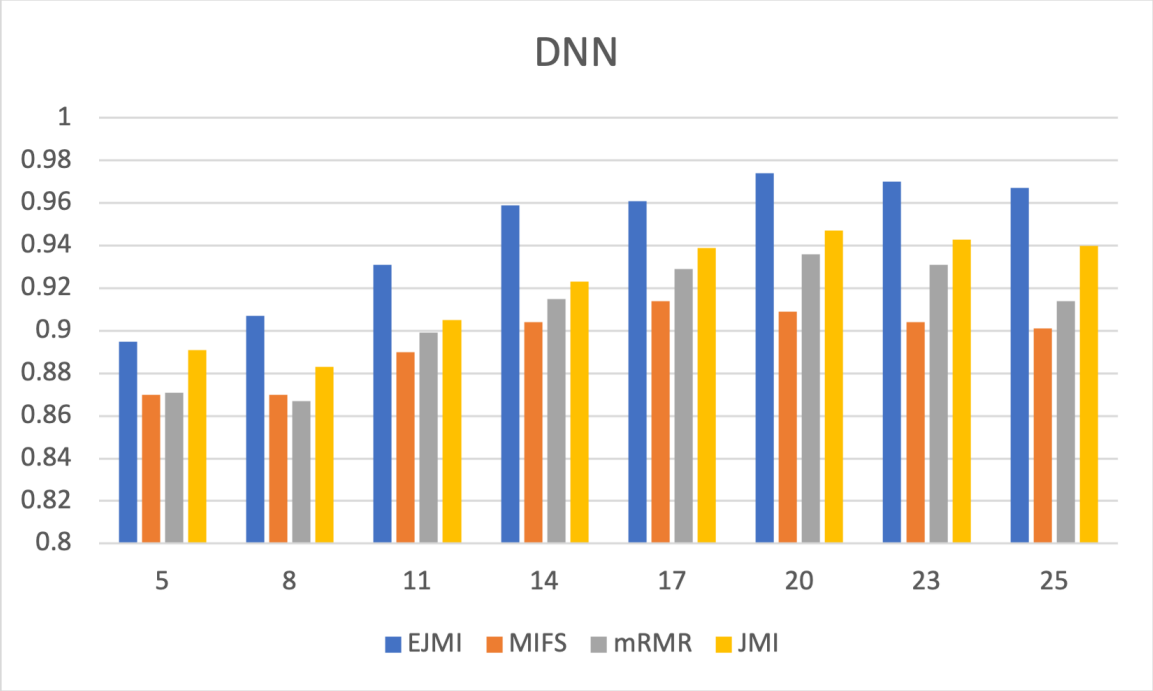

Figure 5.2: Accuracy comparison between the proposed PCRC incorporated into the EJMI for feature selection and the existing feature selection techniques using DNN.

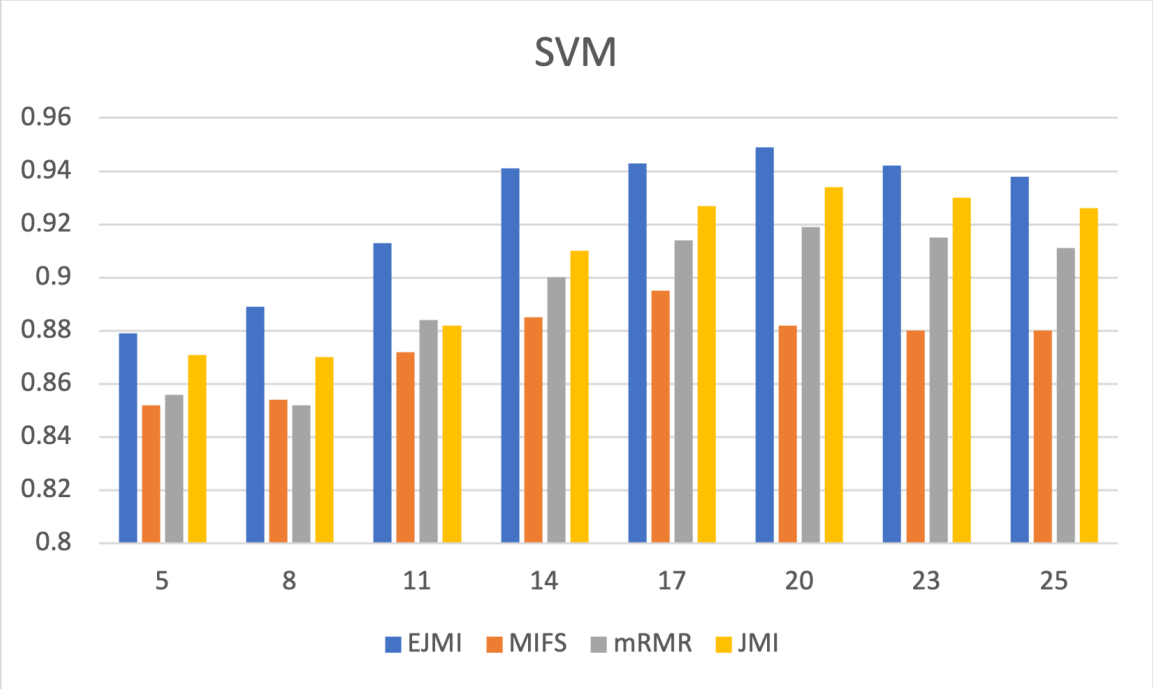

Figure 5.3: Accuracy comparison between the proposed PCRC incorporated into the EJMI for feature selection and the existing feature selection techniques using SVM.

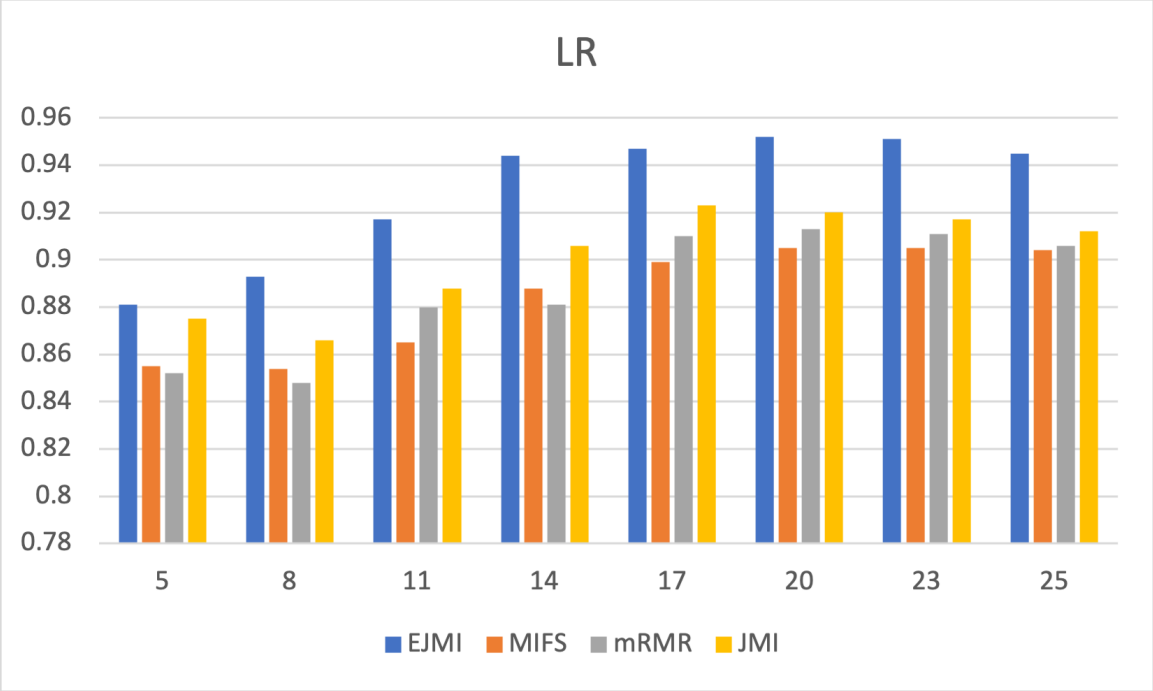

Figure 5.4: Accuracy comparison between the proposed PCRC incorporated into the EJMI for feature selection and the existing feature selection techniques using LR.

5.6 Conclusion

In this chapter, a new Proportional Conditional Redundancy Coefficient (PCRC) technique is proposed. PCRC estimates the value of the conditional redundancy coefficient that helps the EJMI (Enhanced Joint Mutual Information) to more precisely determine each feature's significance. PCRC increases the weight of conditional redundancy terms proportional to the size of the selected feature set. With such an approach, the PCRC addresses the issue of insufficient data captured right after the model is updated in response to the cITS environmental change. Such candid redundancy estimation compensates for the lack of sufficient attack patterns. This mechanism increases the weight of the PCRC more accurately as compared to existing feature selection techniques, and helps to select more relevant features as demonstated in the prior sections.

The proposed technique was evaluated using the NGSIM dataset of cITS, from which

the relevant features were selected. Those features were used to build the IDS using several machine learning and deep learning classifiers. The results obtained show that the conditional redundancy calculated by the proposed PCRC EJMI improved the accuracy of features significance estimated by the EJMI feature selection techniques and reduces redundant features. This, in turn helps in mitigating the model's over fitting by reducing the data dimensionality that adversely affects the generalizability of the detection model and its ability to perceive the hidden patterns from malicious nodes lurking in cITS. By adopting this solution for cITS, driving safety and traffic efficiency can be significantly improved.

Chapter 6

Adaptive IDS for Cooperative Intelligent Transportation Systems Using Deep Belief Networks

This chapter is exactly as published in July 2022 at the (Algorithms) [102]. It addressed the first objective, which is "propose an adaptive deep learning-based misbehavior detection model by integrating the techniques proposed in phases 1 and 2 into a deep belief network structure to increase the ability to detect polymorphic attacks and decrease the false alarms." This model (including all three phases) can adapt to the changes in the cITS environment and cope with the dynamic nature of the attacks. This chapter answers the following question:

Q.3: How can an adaptive approach be utilized to increase and cope with the dynamicity and polymorphic nature of attacks?

6.1 Abstract

The adoption of cooperative intelligent transportation systems (cITSs) improves road safety and traffic efficiency. Vehicles connected to cITS form vehicular ad hoc networks (VANET) to exchange messages (as described above). Like other networks and systems, cITSs are targeted by attackers intent on compromising and disrupting system integrity

and availability. They can repeatedly spoof false information causing bottlenecks, traffic jams and even road accidents. The existing security infrastructure assumes that the network topology and/or attack behavior is static. However, the cITS is inherently dynamic in nature. Moreover, attackers may have the ability and resources to change their behavior continuously. Assuming a static IDS security model for VANETs is not suitable and can lead to low detection accuracy and high false alarms we have developed an adaptive security solution based on deep learning and contextual references that can cope with the dynamic nature of the cITS topologies and increasingly common attack behaviors. The outcome of this approach includes deep belief networks (DBN) used to train the detection model. Binary cross entropy was used as a loss function to measure the prediction error. Two activation functions were used, Relu and Softmax, for input–output mapping. The Relu was used in the hidden layers, while the Sigmoid was used in the last layer to map the real vector to an output between 0 and 1. The adaptation mechanism was incorporated into the detection model using a moving average that monitors predicted values within a time window. In this way, the model can readjust the classification thresholds on-the-fly as appropriate. The proposed model was evaluated using the Next Generation Simulation (NGSIM) dataset as before, which is commonly used in such related works. The result is improved accuracy, demonstrating that the adaptation mechanism used in this study was highly effective

6.2 Introduction

Cooperative intelligent transportation systems (cITSs) collect data from the end nodes (i.e., endpoints). These data are stored locally and shared with the other nodes [40,72,103]. The cITS adopts one of the two information-sharing standards, the European standard [4] and the American standard [4]. On the one hand, the European standard defines two

types of messages, the Cooperative Awareness Message (CAM) and the Decentralized Environmental Notification Message (DENM) [6]. The CAMs are sent periodically and carry information about the vehicles such as their position, size, speed, and the angle of the steering wheel. The DENM messages carry information about events which occur on sections of road such as lane changes and (sudden) braking. On the other hand, the American standard defines context information messages called basic safety messages (BSMs), which carry different information such as position, heading, speed, acceleration, steering angle, vehicle role, vehicle size and status of vehicle lights [1]. If an event happens, then the BSM also carries those event-related information.

Notwithstanding, cITSs enable information sharing among neighboring nodes (i.e., vehicles). Unfortunately, this comes at the cost of needing to address several threats that target data and system integrity [100, 101]. These threats could be imposed by either human-crafted attacks or malware [101, 104–106]. Threats which target cITS systems can disable or disrupt the function of one or more components in the vehicle's navigation system [95].For example, threats can spoof the exchanged data to inject false mobility information which is then exchanged among neighboring vehicles causing erroneous actions and calamitous outcomes.

Threat actors use sophisticated strategies and employ malware to carry out various attacks against cITSs [2, 3]. These attacks could come from nodes inside or outside the network. Outside attacks by threat actors that are not part of the network are easy to detect, whereas inside attacks are usually carried out via legitimate but compromised vehicles. Such inside attacks are more challenging to detect. Typical cITS targeted attacks include jamming, replay, Sybil, and data falsification.

Jamming is carried out by overwhelming individual cITS nodes by an enormous

amount of messages, which disrupt the connectivity with the cITS, a denial-of-service attack type [107]. The consequences include message loss within the cITS, causing a data insufficiency situation that adversely affects the accuracy of the intrusion detection systems (IDS) trained on such data. Replay attacks occur if the attacker can impersonate an original node enabling the interception of messages exchanged between the vehicles and thereby injecting false data by re-sending them to a victim node [108]. Likewise, a Sybil attack creates several identities and uses them to poison (fake) BSM messages that deceive victim nodes; as such, a Sybil attack compromises network services when an attacker subverts the service's reputation system by creating a large number of pseudonymous identities and then using them to gain a disproportionately large influence. Thus, false data injection can be used to share and promote false information about the current traffic situation on the road for the purpose of disrupting traffic flow and triggering congestion.

Data falsification is another type of attack that can be conducted to compromise BSM messages exchanged between cITS nodes. The first step is to compromise a legitimate node and employ it to share false data with neighboring vehicles. Since the compromised node has been previously authenticated, a trust relationship was established with other nodes in the cITS network. Attackers can utilize this fact to spread the false data using the compromised node trust [6]. Attackers thus manipulate the BSM and inject false data which is then share with neighboring nodes [7]. The false data may cause a vehicle to take unexpected actions such as sudden braking, lane changing, and/or sudden acceleration. Therefore, taking security measures to protect BSM messages is crucial [1].

6.3 Related Works

The current solutions proposed for protecting the cITSs can be categorized into node-centric and data-centric IDSs. Some of these solutions tried to protect the system against threats

coming from the outside caused by Sybil, malware, and DoS attacks. By comparing the patterns from incoming traffic with the patterns of normal applications, those solutions can detect suspicious threats and raise alarms. Moreover, other solutions focus on detecting misbehaving nodes in cITSs. These solutions aim to protect the system against threats carried out by legitimate yet compromised nodes, which is more challenging as those nodes are trusted and thus less suspicious [8]. Nonetheless, most of these solutions assume that the cITS is stationary. Such an assumption is not realistic as the ephemeral nature of cITSs make it a very dynamic constantly changing topology. Developing data-driven detection solutions on presumed stationary data prohibits handling the numerous and rapid changes typical inside the cITS. These solutions quickly become outdated and consequently, their accuracy decreases. Some studies have tried to rectify the issue by adopting solutions with the dynamic nature of the operating environment in mind [100]. These solutions, again, are typically categorized into node-centric and data-centric.

The existing IDS proposal for cITS relies on the BSM messages exchanged between the communicating vehicles as well as the contextual metadata that describes the operating environment. Such data in many studies are static, which might not be suitable for dynamic cITSs where the node's operational environment changes continuously. Therefore, static security thresholds become outdated more often. This represents a major issue for existing IDS solutions. To address this issue, some studies have proposed solutions, such as the context-aware data-centric misbehavior detection scheme (CA-DC-MDS) developed by [2]. This solution overcomes the aforementioned drawbacks. Static thresholds are replaced by a dynamic threshold statistically determined using a contextual model, which is constructed and updated online. The sequential analysis of temporal and spatial correlation is conducted using Kalman and Hampel filters to assess the consistency of

mobility data exchanged between neighboring vehicles. The Kalman filter tracks mobility data from the neighboring vehicles, while the Hampel filter assesses the consistency of these data. Based on the proximity from the threshold, the message containing the data is classified as either normal or suspicious. However, the scheme assumes that data collected at the early phases after the model has updated its profile are sufficient for consistency assessment. This is not realistic in most cases, as the contextual data that describe the new situation are not yet ready for a variety of reasons as described below.

Node-centric IDSs determine whether a vehicle is malicious based on how it behaves on the road section [109]. The trustworthiness of legitimate vehicles is also assessed based on such behavior, which can be perceived by observing the number and validity of BSM messages shared by the vehicle [110, 111]. Reputation-based evaluation is usually adopted for the trustworthiness estimation of each node in the cITS. The estimation is performed by a voting strategy whose outcome relies on the majority concept. However, relying on node behavior is sub-optimal because the cITS is non-stationary and since nodes change their behavior as the topology change [112, 113]. Moreover, relying on a voting approach for the trustworthiness estimation is always biased towards the majority, which in some cases, can be compromised when the attacker gains a majority foothold. A case in point occurs when attackers use advanced and sophisticated attack strategies such as malware and botnets to create a majority of rogue nodes enabling them to control the trustworthiness estimation. Consequently, such reputation-based mechanisms used by node-centric solutions cannot be trusted for the early identification of misbehaving or faulty vehicles [1].

Another set of IDSs for cITS adopt the data-centric detection approach by inspecting the BSM messages exchanged between the neighboring vehicles. These solutions perform

several checks to determine whether the messages are falsified. BSM messages are checked against several criteria such as consistency and plausibility to determine whether they are trustworthy [1]. The consistency checks that BSM messages undergo in data-centric solutions determine whether the data shared by the node are consistent with the general context from the particular cITS. By vetting these BSMs, data-centric solutions can also identify the plausibility of the shared data to help in determining validity (i.e., whether they are in-line with those coming from other nodes in the cITS system).

The node-centric and data-centric approaches adopted in existing IDS solutions for cITS rely on estimating the reputation of the nodes and trustworthiness of the data they share with each other. However, both approaches have inherent weaknesses and may not be suitable for tumultuous environments such as cITSs. In such dynamic systems, the nodes join and leave the network frequently, which creates an unstable topology. This makes it difficult to capture sufficient and consistent patterns that represent all behavioral aspects of the nodes. Therefore, existing security solutions with rigid thresholds are not suitable as they do not have the sufficient data needed for accurate decisions. Therefore, these solutions suffer from a high rate of false alarms. Thus, data insufficiency makes it difficult for adaptive mechanisms used by some solutions to accurately calculate the new thresholds, which also have a negative effect on IDS accuracy.

The contribution of this chapter is two-fold:

- A bi-variate moving average (BiMAV) technique was used. Unlike existing methods that only rely on the values estimated at the output layer, BiMAV correlates the changes of the output layer with the averaged input variables. Such an approach provides precise change detection by avoiding the instantaneous changes that could compromise the stability of the detection model.

- Our method was incorporated into the detection model, which helps to prevent the unnecessary re-adjustment of security thresholds at the output layer of the DBN classifier thanks to the BiMAV used to monitor and detect the change in the classification accuracy estimation.

The rest of the chapter is organized as follows. Section 3 presents the methodology in which we describe the proposed solution. The results are analyzed and discussed in Section 4 along with a comparison with existing related work. Section 5 concludes the chapter with a summary of the contribution and findings.

6.4 Methodology

Given the literature reviewed above (Chapter 2), we have concluded that the ephemeral nature of cITSs is a major challenge that makes many existing solutions ineffective. To overcome such a challenge, herein we propose an adaptive IDS for cITS. Our adaptive approach has the ability to cope with the dynamical nature of the cITS operating environment. The BiMAV method was developed to detect the (potential) diversion, in practice, from the existing threshold used by the detection model. Unlike existing methods that rely only on the values estimated at the output layer, BiMAV correlates the change of output layer with the averaged input variables. Such an approach provides precise change detection by avoiding the instantaneous changes that will eventually compromise the stability of the detection model. In this way, the method prevents the unnecessary re-adjustment of security thresholds at the output layer of the DBN classifier thanks to the BiMAV used to monitor and detect the change in the classification accuracy estimation. This is important for dynamic environments such as cITSs where sufficient data might not be available. Based on the amount of change, adaptation can be triggered. In other words, if the difference exceeds a certain limit (i.e., according to the standard deviation),

retraining the model is triggered. Thus, model retraining will be performed based on the new data. If the difference does not exceed the threshold, there is no need for retraining.

Thus, our solution here relies on the supervised learning approach. The deep belief network (DBN), one of the famous deep learning algorithms, is used to train the IDS based on data collected from the BSM messages. Before training, the data are pre-processed to make them suitable for ingestion by the DBN. As part of the preparation, noisy data are removed, and data normalization is carried out. During data normalization, the values of all attributes are converted to a range from 0–1. This ensures that all attributes are in the same scale and prevents those with higher ranges from having undue influence over the model's output decision.

The data are now ready for the mutual information feature selection (MIFS) process that selects out discriminative features to reduce data dimensionality. This avoids the overfitting problem that negatively affects the accuracy of the IDS [63, 114]. By selecting the most relevant features, the model also generates fewer false alarms, which contributes to higher precision. Furthermore, reducing data dimensionality helps decrease the model complexity, which is more favorable for ephemeral environments such as cITSs. The MIFS ranks the features based on the entropy, such that those with higher entropy value correspond to a lower rank. Then, the MIFS selects the n-top ranking features (n experimentally chosen to give higher accuracy). The selected features are then used as input for the DBN algorithm.

During the model's training phase, the DBN is trained using the data and features selected by the MIFS. The DBN model is composed of several layers, namely input, output and hidden. The number of input layer nodes is determined by the number of features selected by the MIFS. These nodes receive data and process them into the hidden layers,

after being scaled (i.e., multiply) by an input weight. In our methodology, the hidden part of the DBN is constructed from three layers. The number of hidden layers is determined based on an overfitting factor during the training phase. The number of nodes in the hidden layer is thus determined based on the bias factor during the training phase. The value of the bias factor was set to 0.25, multiplied by the standard deviation $\sigma(W)$ of the previous window. Therefore, the number of nodes in the hidden layers were taken as a percentage of the original number of layers. As we start with 18 nodes (because the number of nodes in a hidden layer should be lower than the number nodes in input layer), in the hidden layers, the data are processed based on the activation function used by the hidden nodes. The Relu function is used as the activation function in all nodes in the hidden layers of the DBN, except the layer that precedes the output, where the sigmoid function was used. These activation functions are used to map the output of nodes into values between 0 and 1, which are needed for prediction. The output layer receives the data from the sigmoid functions in the last hidden layer and determines whether the instance is malicious or normal based on a threshold σ, where values greater than σ are considered as attacks.

6.4.1 Training and Testing

The DBN model was trained using the 10-fold cross-validation method, wherein data are divided into two sets. During the training/testing process, the data were divided into aforementioned two sets, i.e., training and testing. The training builds the model while testing evaluates its accuracy. The size of the training set was 90% of the data and, naturally, the testing set was 10% of the data. This process was repeated 10 times and the accuracy of the model was recorded. At the end of the training/testing process, the averaged accuracy was calculated, which determines the overall model accuracy.

6.5 Model Adaptation using Bi-Variate Moving Average

Our proposed model, as described above, is aimed at improving detection within the dynamic cITS environment. Therefore, here we describe an adaptation capability needed to ensure that the model can better handle the constantly changing network topology. We propose a bi-variate moving average (BiMAV) model adaptation method that observes the model performance and adapts to the change in the operating environment. Our method follows the progressive modeling used by works that rely on time series data [115]. The method uses a two-dimensional window for change detection. That is, the window defines two variables, the aggregated input values and the estimated output. Within this window, the accuracy trend is monitored against a threshold calculated based on the standard deviation from previous windows. Equation (6.1) implements the BiVAM method:

$$BiMAV = \frac{\sum_{i=0}^{i=n-1} X_i}{n} \times \frac{\sum_{j=0}^{j=l-1} Y_j}{l} \tag{6.1}$$

where X_i and Y_j are the input features and estimated output values, respectively. The variable n represents the number of features while l represents the number of instances in the window. The retraining is triggered if the value of BiMAV is higher than the standard deviation of the previous windows, as expressed by Equation (6.2):

$$\text{B i M A V} = \begin{cases} if < \sigma(W) \\ if > \sigma(W) \quad \text{then Retraining} \end{cases} \qquad \text{then No retraining} \tag{6.2}$$

where $\sigma(W)$ represents the standard deviation of the previous windows. The decision that Equation (2) makes is binary as it determines whether the re-training is needed or not based on the threshold $\sigma(W)$.

6.6 The Dataset

The dataset used for this study was the Next Generation Simulation (NGSIM) Vehicle Trajectories Dataset [101]. NGSIM is an open source publicly available dataset with a collection of real-world vehicles' trajectories collected by smart vehicles. It contains a detailed vehicle trajectory data on southbound US 101 and Lankershim Boulevard in Los Angeles, CA, eastbound I-80 in Emeryville, CA and Peachtree Street in Atlanta, Georgia. Data in NGSIM were collected through a network of synchronized digital video cameras. NGVIDEO, a customized software application developed for the NGSIM program, transcribed the vehicle trajectory data from the video. This vehicle trajectory data provides the precise location of each vehicle within the study area every one-tenth of a second, resulting in detailed lane positions and locations relative to other vehicles. Moreover, NGSIM consists of many patterns representing different driving situations and driver behavior [101]. In addition, NGSIM provides high-quality contextual data that describe realistic real-world scenarios on different road sections [109]. Particularly, NGSIM was built by collecting data from vehicles moving on a road section 500 meters long and seven-lanes of highway wide. For each vehicle, the data are collected (recorded) for 45 min using 16 sensors. Each record in the dataset contains s set of basic elements regarding the vehicle position, speed, time, direction, and acceleration. Although there are similar datasets such as the Connected Vehicles Pilot (CVP), the NGSIM dataset was chosen in here to be consistent when comparing with the related works as they also used the NGSIM.

The dataset represents the ground truth information and each vehicle represents a cITS node. In a real-world deployment, the dataset needs to be fed each cITS node. That is, each node should have a copy of the dataset to run its own applications and adjust

its communication or driving behavior. As such, the collection of accurate and reliable context information is crucial. The context information in the dataset combines two types of messages, cooperative awareness message (CAM) and decentralized environmental notification messages (DENM) into a basic safety messages (BSM). While CAMs are sent periodically, DENMs are event-driven that only get sent when an event has occurred. The CAM consists of information about the vehicles such as the position, size, speed, and steering angle.

In contrast, DENM contains information about a certain event such as lane changing and sudden braking. BSM combines CAM and DENM messages. The first part of BSM, as well as CAM in the European standard, carries information about position, heading, speed, acceleration, steering angle, vehicle role, vehicle size, and the status of vehicle lights [4, 33, 116]. Unlike the first part of BSM that is included in all BSM messages, the second part of BSM (which corresponds to the DENM in the European standard) is only included when an event happens, to carry information about such an event.

6.7 Experimental Environment Setup

To implement the different components of the proposed mode and evaluate its performance, the development and experimental evaluation will be conducted using several tools and software including Python, TensorFlow, Scikit Learn, SKFeature, and Numpy. These tools and libraries are all included in the Anaconda development platform. Meanwhile, the preparation of data samples, implementation of algorithms, and the analysis of the results will be carried out on a machine with Intel(R) Core (TM) i7-4790 CPU @ 3.60 GHZ and 16 GB RAM.

6.7.1 Evaluation Metrics

The evaluation metrics used here can be seen in Chapter 3 (Section 3.6).

6.8 Experimental Results

Table 6.1 shows the accuracy (Acc), detection rate (DR), false positive rate (FPR), and F1 measure of the proposed Adaptive Deep Belief Network-Based IDS (ADBN-IDS). In addition, Tables 6.2 and 6.3 show the results of the IDS built using conventional machine learning classifiers, namely the support vector machines (SVMs), and the logistic regression (LR). As pointed out previously in Chapter 5, the Acc, DR, FPR, and F1 were calculated based on Equations 3.1, 3.2, 3.4, and 3.5. In the tables, the first column in each table lists the accuracy of the proposed; while the second lists the detection rat; the third column lists the false positive rate; and the fourth column lists the F1 measure of the proposed and related models. The tables' rows are used to list the feature sets with different sizes. The feature sizes range between 5 and 25 incremented by 3. The results show that the proposed ADBN-IDS achieved higher accuracy over the other two classifiers (i.e., SVM and LR) [116, 117]. This is attributed to the ability of the BiMAV method (incorporated into ADBN-IDS) to detect the degradation in the model's performance and trigger the training on the right time. This contributes to keeping the model up to date and prevent the concept drift from affecting the accuracy of the model. Notwithstanding, these experiments duplicate (repeat) the prior experiments carried out in the earlier phases.

The results also show that the accuracy increased when more features were added, until the number of features reached 20. After that, the model experienced a decrease in the accuracy. This also can be observed from the other evaluation metrics, namely

DR, FPR, and F1. The same trend was observed not only for the ADBN-IDS, but also for SVM and LR. The reason is that the model needs sufficient features to make correct decisions. However, when the number of features exceed a certain limit, the model would suffer from high variance that makes it prone to overfitting. The situation exacerbates when the coming observations lack the sufficient attack patterns necessary for clear and accurate decisions. This would result in a model that can only recognize the patterns that it has seen, and if new patterns that have less similarity with the known ones are encountered, the likelihood that the model could miss the true classification becomes high, just as described in the prior chapters.

Figures 6.1, 6.2, 6.3, and 6.4 show the comparison between the proposed ADBN-IDS and the models built using the SVM and LR, in terms of accuracy, detection rate, false positive rate, and F measure, respectively. The x axis represents the number of features used for training, and the y axis represents the value of performance measure achieved. The comparison was conducted between the ADBN-IDS that employed the BiMAV for adaptation and the conventional approach used in the existing studies [116, 117]. As depicted in the figures, the proposed ADBN-IDS outperformed the related techniques in terms of accuracy, detection rate, false positive rate, and the F measure. Note, it can also be observed that the ADBN-IDS maintain a stable increment in the performance for the four measures when the number of features increase until it reaches 20 features where the performance shows a declining trend. This is attributed to the efficacy of the BiMAV incorporated for the model adaptation and the reliance on the combination of output and averaged inputs for proximity calculation from the threshold. Such an approach makes the change detection mechanism robust, which avoids unnecessary re-training and only triggers it if the change in the cITS topology or attack behavior is significant. Let us also

Table 6.1: The experimental evaluation results for the proposed ADBN-IDS in terms of accuracy, detection rate, false positive rate, and F measure.

Metric and Number of Features	**ACC**	**DR**	**FPR**	**F1**
5	0.92	0.924	0.132	0.927
8	0.929	0.926	0.128	0.931
11	0.946	0.937	0.113	0.947
14	0.968	0.965	0.084	0.969
17	0.97	0.968	0.076	0.973
20	0.974	0.972	0.071	0.978
23	0.973	0.97	0.072	0.975
25	0.969	0.971	0.077	0.972

Table 6.2: The experimental evaluation results for the proposed SVM-IDS in terms of accuracy, detection rate, false positive rate, and F measure.

Metric and Number of Features	**ACC**	**DR**	**FPR**	**F1**
5	0.892	0.89	0.176	0.894
8	0.9	0.894	0.179	0.892
11	0.91	0.913	0.15	0.915
14	0.951	0.95	0.132	0.954
17	0.956	0.953	0.129	0.958
20	0.957	0.953	0.122	0.958
23	0.951	0.948	0.13	0.953
25	0.947	0.942	0.154	0.951

note that the frequency of adaptation varies based on the threshold. When the threshold is set to a higher value, the rate of adaptation becomes less frequent. When the threshold value is set to low, the adaptation frequency increases. Moreover, Figure 6.5 shows the area under the curve of the proposed model given several thresholds as shown. The x axis represents the false positive rate while the y axis represents the true positive rate. It can be observed that the false positive rate decreases when the detection rate increases.

Table 6.3: The experimental evaluation results for the proposed LR-IDS in terms of accuracy, detection rate, false positive rate, and F measure.

Metric and Number of Features	ACC	DR	FPR	F1
5	0.898	0.894	0.162	0.9
8	0.904	0.902	0.157	0.907
11	0.919	0.917	0.144	0.918
14	0.943	0.94	0.14	0.946
17	0.958	0.952	0.131	0.96
20	0.954	0.951	0.137	0.956
23	0.95	0.948	0.139	0.952
25	0.945	0.943	0.142	0.948

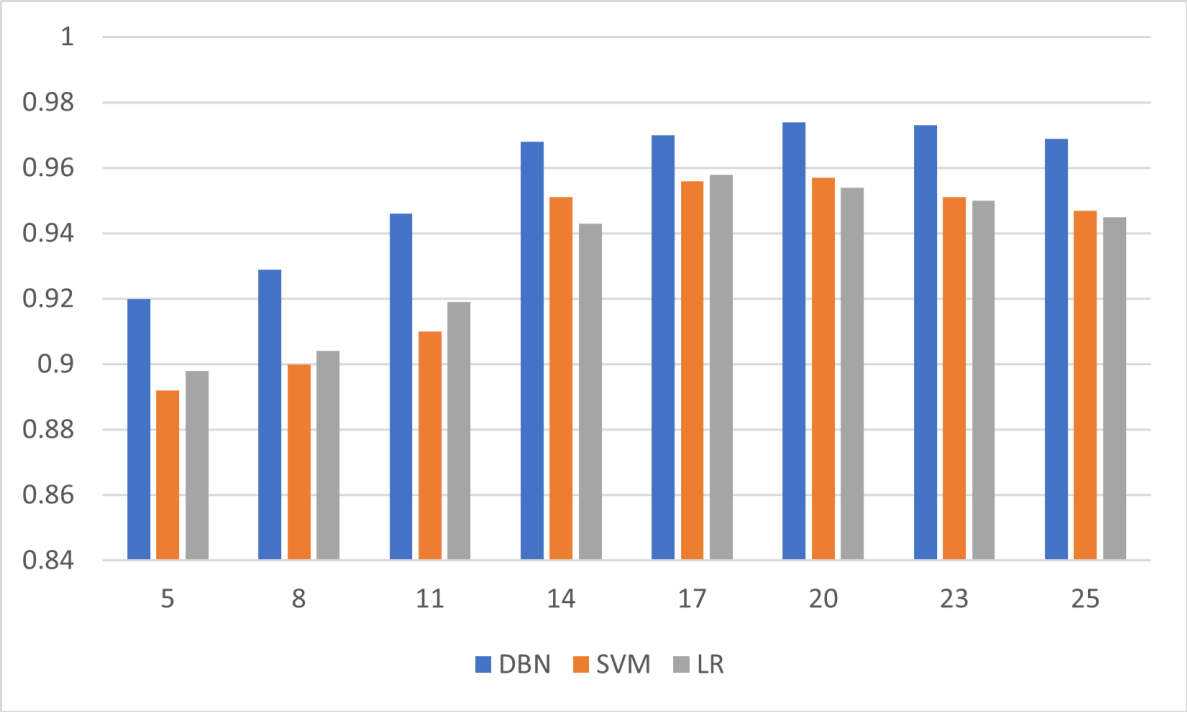

Figure 6.1: Comparison of the proposed ADBN-IDS with SVM and LR in terms of detection accuracy.

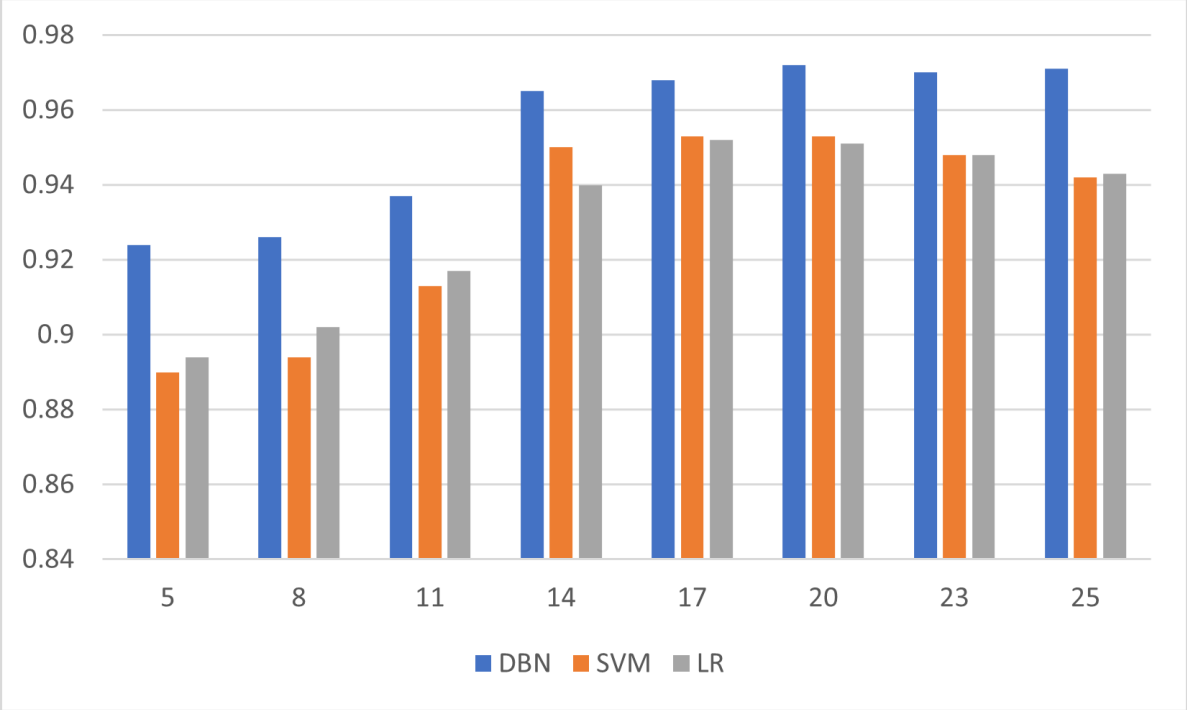

Figure 6.2: Comparison of the proposed ADBN-IDS with SVM and LR in terms of detection rate.

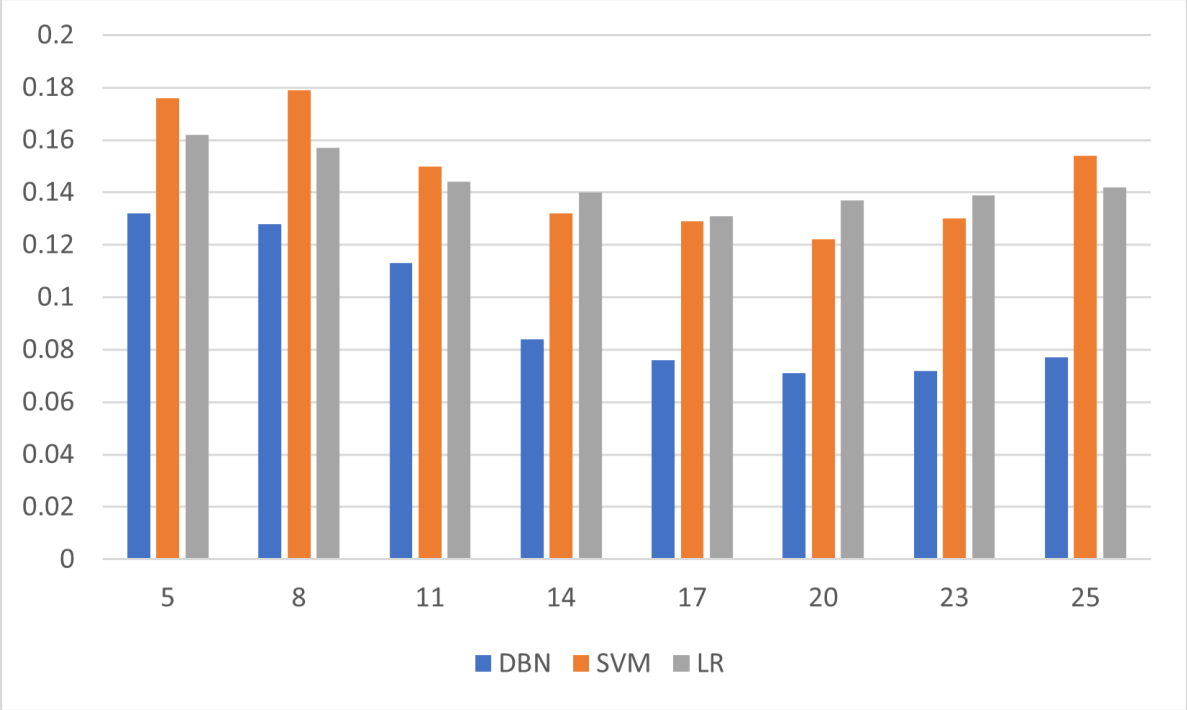

Figure 6.3: Comparison of the proposed ADBN-IDS with SVM and LR in terms of false positive rate.

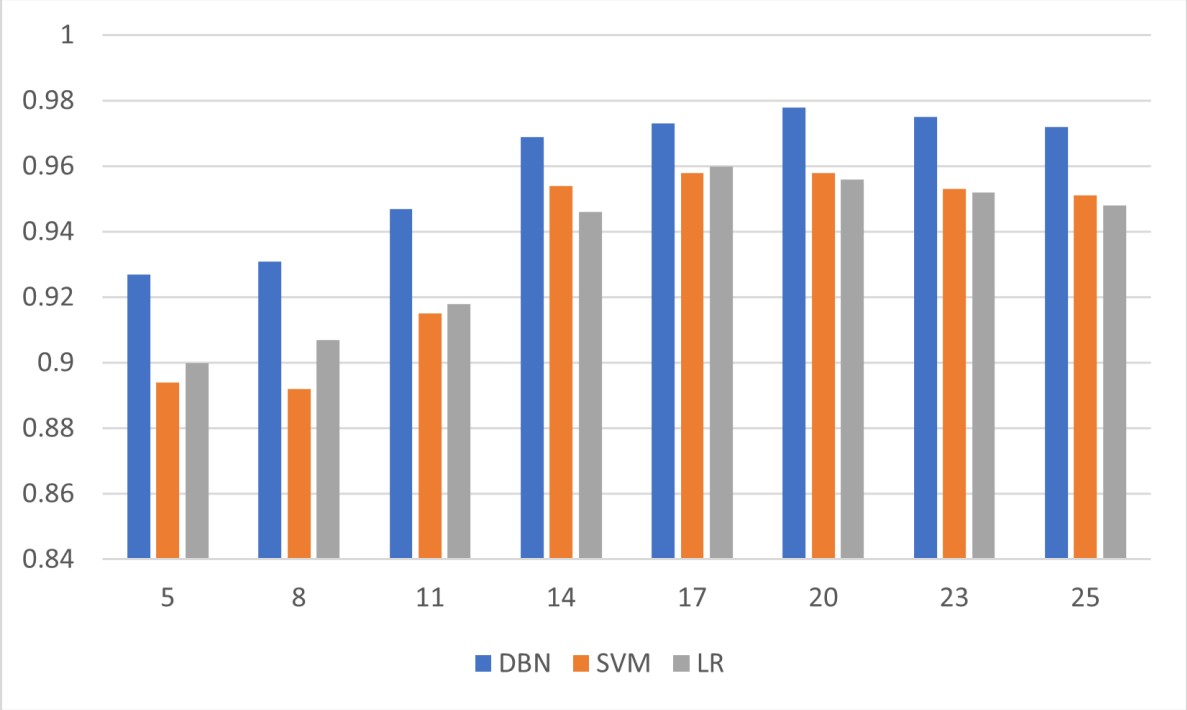

Figure 6.4: Comparison of the proposed ADBN-IDS with SVM and LR in terms of F measure.

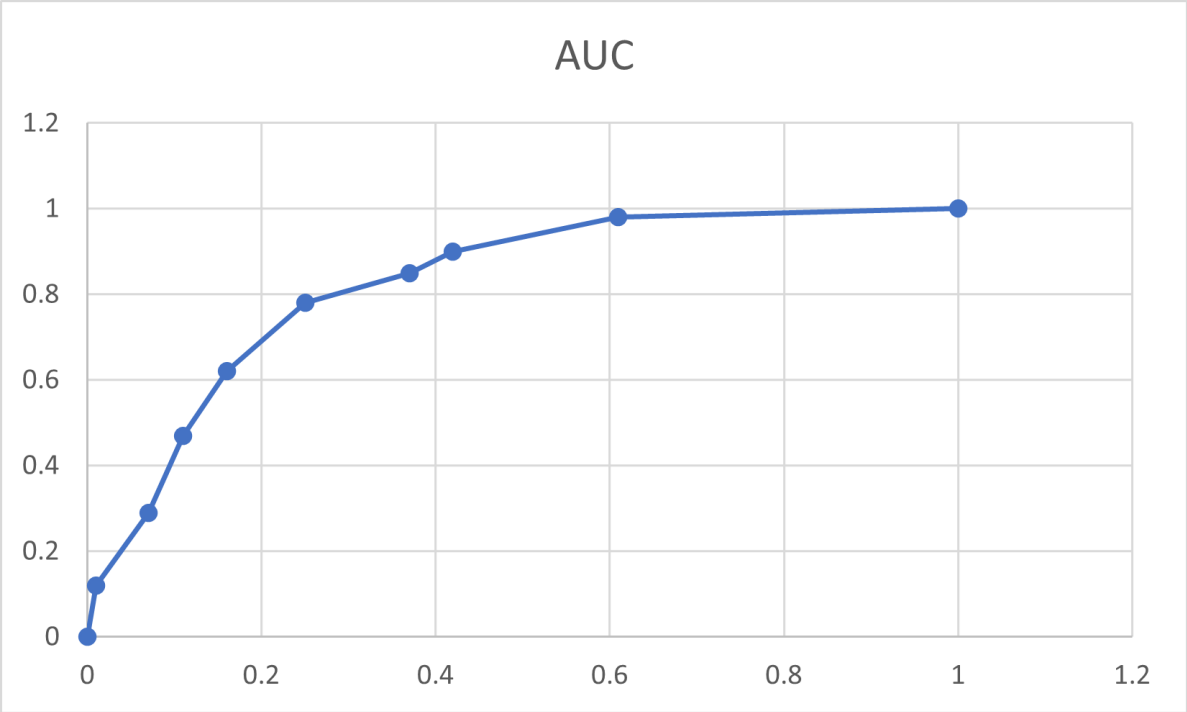

Figure 6.5: Area under the curve comparison for several detection thresholds.

6.9 Conclusions and Summary

In this chapter, our ADBN-IDS for cITS is described. ADBN-IDS is composed of three components: pre-processing, feature selection, and training/testing. Thus, the model is created from the deep belief network (DBN) classifier, and includes the bi-variate moving average (BiMAV) method as our adaptation technique. This inclusion allows the model to cope with the dynamic nature of the cITS environment and has never been tested using the NGSIM dataset.

The classifier was trained using the NGSIM dataset and tested using 10-fold cross validation. The performance of the model was evaluated using several metrics including accuracy, detection rate, false positive rate, and the F1 measure. The evaluation of our results demonstrate that the proposed ADBN-IDS achieved higher performance in terms of accuracy, detection rate, false positive rate, and F1, which indicates the importance of the BiMAV adaptation mechanism in achieving and maintaining a safer more resilient cITS.

In summary, our new ABDN-IDS model, for the NGSIM dataset, showed on average, an improvement of 2.35%, 2.47%, and 42% in terms of accuracy, detection and false positive rate, respectively.

Chapter 7

Summary, Conclusion and Future Work

This dissertation focused on building an adaptive IDS that can detect the evasive attacks against the cITSs. Throughout this research study, the achievement of the three objectives mentioned in Chapter 1, Section 1.4 were carried out in phases. In phase 1, the first objective was achieved through the development of a local-global fuzzy clustering estimation method in which the missing values are imputed. Again, our estimation which is implemented based on the surrounding values of the same, as well as, correlated attributes. The results show that the proposed method achieves a significant improvement as compared to existing techniques reported in the literature. There is one exception, namely the FPR derived from the CNN model (i.e., -59%). As can be seen in Table 7.1 which reports the percentage improvement for Phase I, our results compared to existing techniques are all better with only that one exception. The estimation was implemented based on the surrounding values of the same as well as correlated attributes. The results show that the proposed method achieved improvement compared to the existing techniques except for the FPR of CNN where the existing technique was the lower (see Table 7.1).

The 2nd objective develops our Proportional Conditional Redundancy Coefficient

Table 7.1: Phase 1 percentage improvement compared to the existing techniques (refer to Section 4.4, Table 4.1)).

#Metric / Classifier	LR	SVM	CNN
ACC	4.4%	3.8%	3.6%
F1	2.9%	2.2%	0.8%
FPR	47%	46%	-59%
DR	7.40%	5.90%	5.40%

(PCRC) which also enhances the feature selection process because of the improved feature significance estimation which takes place during the Phase 2. Consequently we achieved the following percentage improvements, namely that Accuracy (ACC) was improved 2.6% better than reported using Deep Neural Network (DNN), 2.0% better than using Support Vector Machines (SVM) and the improvement was 3.1% using Logistic Regression (LR).

Finally, the third objective was achieved within Phase 3 by incorporating our Bi-variate Moving Average (BiMAV) technique into the DBN-based detection model and adapting to the changes on-the-fly within the cITS system. In summary, the results show generally an overall improvement. Our method achieved improvements as compared to existing techniques as follows: ACC provided a 2.4% improvement compared to SVM, Detection Rate (DR) provided 2.5% improvement compared to SVM, False Positive Rate (FPR) provided a 42% improvement compared to SVM, and the F1 improved 2.4% over that provided by those published using SVM.

To recap, our model's success was derived using several statistical and machine learning algorithms. We began the research study journey by evaluating several strategies for discovering the best methods of misbehavior detection in cITS environments. In the end, we developed a i) technique for missing values imputation and feature extraction. Then,

we developed an ii) improved redundancy coefficient estimation method for better feature selection. This method improves the iii) process of identifying the relevant features that are later used to train the adaptive deep learning-based model in Phase 3 (Chapter 6). Ultimately, we evaluated and compared our results for those techniques and methods using several common metrics (i.e., accuracy, false positive rate, and detection rate) used in similar research.

7.1 Future Work

Below, the first section, 7.1.1, describes the need for traffic data diversity. The second section, 7.1.2, is concerned with gaining real-world attack diversity. The third section, 7.1.3, identifies future work concerning the model's computational efficiency.

7.1.1 Data diversity

In data-driven modelling, data is the cornerstone, which dictates the performance of the trained model. At the core of successful training come the ability to introspect different aspects and identify diverse patterns that help the model successfully generalize across a large swath of behaviors. As the NGSIM dataset used in this research is taken from a single area, the patterns that modelling techniques can explore are limited. This adversely affect the applicability and suitability of our model for other areas or slightly different environments. This is a major challenge that can be addressed by adopting ensemble learning in a way that combines multiple different models into a one generic approach that can be easily adapted to the specific environment on-the-fly as needed.This goal will take a great deal of diversity in the data which may require additional dataset captures that provide the ground truth for a wide range of different scenarios and behaviors.

7.1.2 Attack Diversity

Attack diversity for cITS-related IDS models is another aspect that can be investigated. In this research, we focused on the attacks that affect the data (e.g., false data injection). However, there are several ways attacks could target the entities causing denial of service (e.g., jamming). In addition, some attacks like ransomware could lock the services or the data, which might lead to catastrophic system failures. These attacks should be investigated in the context of cITS.

7.1.3 Multiple Sources

In Chapters 4, 5, and 6 we evaluated the performance of our approach from an accuracy standpoint. We used several metrics like accuracy, detection rate, and false positive rate. Although this is common in such studies, it is also recommended to evaluate the efficiency of the proposed model using real-world constraints. Since some nodes in the cITS environment are resource-constrained, i.e., small memory capacity or processing power as well as structural communications interference. Running resource-hungry models like deep learning, might degrade the response time and affect the efficiency of the entire process and system.

Therefore, there is a need for lightweight solutions that will make use of such advanced modelling while maintaining a lower computational footprint on the cITS systems of the future. In this regard, reducing model complexity could significantly impact the development of lightweight solutions and should be investigated as well.

Bibliography

[1] R. W. van der Heijden, S. Dietzel, T. Leinmüller, and F. Kargl, "Survey on misbehavior detection in cooperative intelligent transportation systems," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 779–811, 2018.

[2] F. A. Ghaleb, M. A. Maarof, A. Zainal, M. A. Rassam, F. Saeed, and M. Alsaedi, "Context-aware data-centric misbehaviour detection scheme for vehicular ad hoc networks using sequential analysis of the temporal and spatial correlation of the consistency between the cooperative awareness messages," *Vehicular Communications*, vol. 20, p. 100186, 2019.

[3] F. A. Ghaleb, M. A. Maarof, A. Zainal, B. A. S. Al-Rimy, F. Saeed, and T. Al-Hadhrami, "Hybrid and multifaceted context-aware misbehavior detection model for vehicular ad hoc network," *IEEE Access*, vol. 7, pp. 159 119–159 140, 2019.

[4] "Etsi, t., intelligent transport systems (its); vehicular communications; geonetworking; part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; sub-part 2: Media-dependent functionalities for its-g5. etsi ts," vol. 102, pp. p. 636–4, 2013.

[5] H. Amirat, N. Lagraa, C. A. Kerrach, and Y. Ouinten, "Fuzzy clustering for misbehaviour detection in vanet," in *2018 International Conference on Smart*

Communications in Network Technologies (SaCoNeT). IEEE, 2018, pp. 200–204.

[6] F. A. Ghaleb, M. A. Maarof, A. Zainal, B. A. S. Al-rimy, A. Alsaeedi, and W. Boulila, "Ensemble-based hybrid context-aware misbehavior detection model for vehicular ad hoc network," *Remote Sensing*, vol. 11, no. 23, p. 2852, 2019.

[7] F. A. Ghaleb, A. Zainal, M. A. Rassam, and F. Mohammed, "An effective misbehavior detection model using artificial neural network for vehicular ad hoc network applications," in *2017 IEEE conference on application, information and network security (AINS)*. IEEE, 2017, pp. 13–18.

[8] T. Pandiangan, I. Bali, and A. Silalahi, "Early lung cancer detection using artificial neural network," *Atom Indonesia*, vol. 45, no. 1, pp. 9–15, 2019.

[9] H. Hasrouny, A. E. Samhat, C. Bassil, and A. Laouiti, "Misbehavior detection and efficient revocation within vanet," *Journal of information security and applications*, vol. 46, pp. 193–209, 2019.

[10] J. Zacharias and S. Fröschle, "Misbehavior detection system in vanets using local traffic density," in *2018 IEEE Vehicular Networking Conference (VNC)*. IEEE, 2018, pp. 1–4.

[11] P. K. Singh, S. Gupta, R. Vashistha, S. K. Nandi, and S. Nandi, "Machine learning based approach to detect position falsification attack in vanets," in *International Conference on Security & Privacy*. Springer, 2019, pp. 166–178.

[12] S. Rakhi and K. Shobha, "Performance analysis of an efficient data-centric misbehavior detection technique for vehicular networks," in *International Conference*

on Computer Networks and Communication Technologies. Springer, 2019, pp. 321–331.

[13] H. Tan, Z. Gui, and I. Chung, "A secure and efficient certificateless authentication scheme with unsupervised anomaly detection in vanets," *IEEE Access*, vol. 6, pp. 74 260–74 276, 2018.

[14] L. Nie, H. Wang, S. Gong, Z. Ning, M. S. Obaidat, and K.-F. Hsiao, "Anomaly detection based on spatio-temporal and sparse features of network traffic in vanets," in *2019 IEEE global communications conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[15] L. Nie, Y. Li, and X. Kong, "Spatio-temporal network traffic estimation and anomaly detection based on convolutional neural network in vehicular ad-hoc networks," *IEEE Access*, vol. 6, pp. 40 168–40 176, 2018.

[16] B. A. S. Al-rimy, M. A. Maarof, and S. Z. M. Shaid, "Ransomware threat success factors, taxonomy, and countermeasures: A survey and research directions," *Computers & Security*, vol. 74, pp. 144–166, 2018.

[17] A. Narayanan, M. Chandramohan, L. Chen, and Y. Liu, "Context-aware, adaptive, and scalable android malware detection through online learning," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 1, no. 3, pp. 157–175, 2017.

[18] H. Darabian, A. Dehghantanha, S. Hashemi, S. Homayoun, and K.-K. R. Choo, "An opcode-based technique for polymorphic internet of things malware detection," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 6, p. e5173, 2020.

[19] M. Aloqaily, S. Otoum, I. Al Ridhawi, and Y. Jararweh, "An intrusion detection system for connected vehicles in smart cities," *Ad Hoc Networks*, vol. 90, p. 101842, 2019.

[20] N. Kumar and N. Chilamkurti, "Collaborative trust aware intelligent intrusion detection in vanets," *Computers & Electrical Engineering*, vol. 40, no. 6, pp. 1981–1996, 2014.

[21] H. Sedjelmaci and S. M. Senouci, "An accurate and efficient collaborative intrusion detection framework to secure vehicular networks," *Computers & Electrical Engineering*, vol. 43, pp. 33–47, 2015.

[22] B. Subba, S. Biswas, and S. Karmakar, "A game theory based multi layered intrusion detection framework for vanet," *Future Generation Computer Systems*, vol. 82, pp. 12–28, 2018.

[23] T. Zhang and Q. Zhu, "Distributed privacy-preserving collaborative intrusion detection systems for vanets," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 4, no. 1, pp. 148–161, 2018.

[24] X. Zhang, C. Lyu, Z. Shi, D. Li, N. N. Xiong, and C.-H. Chi, "Reliable multiservice delivery in fog-enabled vanets: Integrated misbehavior detection and tolerance," *IEEE Access*, vol. 7, pp. 95 762–95 778, 2019.

[25] C. Zhang, K. Chen, X. Zeng, and X. Xue, "Misbehavior detection based on support vector machine and dempster-shafer theory of evidence in vanets," *IEEE Access*, vol. 6, pp. 59 860–59 870, 2018.

[26] K. Sharshembiev, S.-M. Yoo, E. Elmahdi, Y.-K. Kim, and G.-H. Jeong, "Fail-safe mechanism using entropy based misbehavior classification and detection in vehicular ad hoc networks," in *2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. IEEE, 2019, pp. 123–128.

[27] F. A. Aboaoja, A. Zainal, F. A. Ghaleb, and B. A. S. Al-rimy, "Toward an ensemble behavioral-based early evasive malware detection framework," in *2021 International Conference on Data Science and Its Applications (ICoDSA)*. IEEE, 2021, pp. 181–186.

[28] F. A. Ghaleb, M. A. Maarof, A. Zainal, M. A. Rassam, F. Saeed, and M. Alsaedi, "Context-aware data-centric misbehaviour detection scheme for vehicular ad hoc networks using sequential analysis of the temporal and spatial correlation of the consistency between the cooperative awareness messages," *Vehicular Communications*, vol. 20, p. 100186, 2019.

[29] X.-Y. Lu and A. Skabardonis, "Freeway traffic shockwave analysis: exploring the ngsim trajectory data," in *86th Annual Meeting of the Transportation Research Board, Washington, DC*. Citeseer, 2007.

[30] B. Coifman and L. Li, "A critical evaluation of the next generation simulation (ngsim) vehicle trajectory dataset," *Transportation Research Part B: Methodological*, vol. 105, pp. 362–377, 2017.

[31] S. A. Almalki and F. T. Sheldon, "Deep learning to improve false data injection attack detection in cooperative intelligent transportation systems," in *2021 IEEE*

12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). IEEE, 2021, pp. 1016–1021.

[32] H. Bangui, M. Ge, and B. Buhnova, "A hybrid machine learning model for intrusion detection in vanet," *Computing*, pp. 1–29, 2021.

[33] F. A. Ghaleb, F. Saeed, M. Al-Sarem, B. Ali Saleh Al-rimy, W. Boulila, A. Eljialy, K. Aloufi, and M. Alazab, "Misbehavior-aware on-demand collaborative intrusion detection system using distributed ensemble learning for vanet," *Electronics*, vol. 9, no. 9, p. 1411, 2020.

[34] P. K. Singh, S. K. Nandi, and S. Nandi, "A tutorial survey on vehicular communication state of the art, and future research directions," *Vehicular Communications*, vol. 18, p. 100164, 2019.

[35] S. Sumithra and R. Vadivel, "Ensemble miscellaneous classifiers based misbehavior detection model for vehicular ad-hoc network security."

[36] A. Talpur and M. Gurusamy, "Machine learning for security in vehicular networks: A comprehensive survey," *arXiv preprint arXiv:2105.15035*, 2021.

[37] M. Lee and T. Atkison, "Vanet applications: Past, present, and future," *Vehicular Communications*, vol. 28, p. 100310, 2021.

[38] R. S. Vitalkar, S. S. Thorat, and D. V. Rojatkar, "Intrusion detection for vehicular ad hoc network based on deep belief network," in *Computer Networks and Inventive Communication Technologies*. Springer, 2022, pp. 853–865.

[39] S. So, P. Sharma, and J. Petit, "Integrating plausibility checks and machine learning for misbehavior detection in vanet," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2018, pp. 564–571.

[40] S. A. Almalki and J. Song, "A review on data falsification-based attacks in cooperative intelligent transportation systems," *International Journal of Computer Science and Security (IJCSS)*, vol. 14, no. 2, p. 22, 2020.

[41] S. Kaffash, A. T. Nguyen, and J. Zhu, "Big data algorithms and applications in intelligent transportation system: A review and bibliometric analysis," *International Journal of Production Economics*, vol. 231, p. 107868, 2021.

[42] Y. A. Ahmed, B. Koçer, S. Huda, B. A. S. Al-rimy, and M. M. Hassan, "A system call refinement-based enhanced minimum redundancy maximum relevance method for ransomware early detection," *Journal of Network and Computer Applications*, vol. 167, p. 102753, 2020.

[43] B. A. S. Al-Rimy, M. A. Maarof, M. Alazab, F. Alsolami, S. Z. M. Shaid, F. A. Ghaleb, T. Al-Hadhrami, and A. M. Ali, "A pseudo feedback-based annotated tf-idf technique for dynamic crypto-ransomware pre-encryption boundary delineation and features extraction," *IEEE Access*, vol. 8, pp. 140 586–140 598, 2020.

[44] A. Saidi, K. Benahmed, and N. Seddiki, "Secure cluster head election algorithm and misbehavior detection approach based on trust management technique for clustered wireless sensor networks," *Ad Hoc Networks*, vol. 106, p. 102215, 2020.

[45] F. A. Ghaleb, B. A. S. Al-rimy, M. Kamat, M. Rohani, S. A. Razak *et al.*, "Fairness-oriented semi-chaotic genetic algorithm-based channel assignment

technique for nodes starvation problem in wireless mesh network," *arXiv preprint arXiv:2006.09655*, 2020.

[46] S. A. Kashinath, S. A. Mostafa, A. Mustapha, H. Mahdin, D. Lim, M. A. Mahmoud, M. A. Mohammed, B. A. S. Al-Rimy, M. F. M. Fudzee, and T. J. Yang, "Review of data fusion methods for real-time and multi-sensor traffic flow analysis," *IEEE Access*, 2021.

[47] L. Deng, X.-Y. Liu, H. Zheng, X. Feng, and Y. Chen, "Graph spectral regularized tensor completion for traffic data imputation," *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[48] R. van der Heijden, S. Dietzel, and F. Kargl, "Misbehavior detection in vehicular ad-hoc networks," *1st GI/ITG KuVS Fachgespräch Inter-Vehicle Communication. University of Innsbruck*, pp. 23–25, 2013.

[49] N. Bißmeyer, "Misbehavior detection and attacker identification in vehicular ad-hoc networks," 2014.

[50] U. Khan, S. Agrawal, and S. Silakari, "A detailed survey on misbehavior node detection techniques in vehicular ad hoc networks," in *Information systems design and intelligent applications*. Springer, 2015, pp. 11–19.

[51] R. W. Van der Heijden, F. Kargl, O. M. Abu-Sharkh *et al.*, "Enhanced position verification for vanets using subjective logic," in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*. IEEE, 2016, pp. 1–7.

[52] F. Sakiz and S. Sen, "A survey of attacks and detection mechanisms on intelligent transportation systems: Vanets and iov," *Ad Hoc Networks*, vol. 61, pp. 33–50, 2017.

[53] O. A. Wahab, A. Mourad, H. Otrok, and J. Bentahar, "Ceap: Svm-based intelligent detection model for clustered vehicular ad hoc networks," *Expert Systems with Applications*, vol. 50, pp. 40–54, 2016.

[54] S. Jamali and R. Fotohi, "Dawa: Defending against wormhole attack in manets by using fuzzy logic and artificial immune system," *the Journal of Supercomputing*, vol. 73, no. 12, pp. 5173–5196, 2017.

[55] N. Bißmeyer, K. H. Schröder, J. Petit, S. Mauthofer, and K. M. Bayarou, "Short paper: Experimental analysis of misbehavior detection and prevention in vanets," in *2013 IEEE Vehicular Networking Conference*. IEEE, 2013, pp. 198–201.

[56] B. Al-Otaibi, N. Al-Nabhan, and Y. Tian, "Privacy-preserving vehicular rogue node detection scheme for fog computing," *Sensors*, vol. 19, no. 4, p. 965, 2019.

[57] K. Zaidi, M. B. Milojevic, V. Rakocevic, A. Nallanathan, and M. Rajarajan, "Host-based intrusion detection for vanets: a statistical approach to rogue node detection," *IEEE transactions on vehicular technology*, vol. 65, no. 8, pp. 6703–6714, 2015.

[58] A.-S. K. Pathan, *Security of self-organizing networks: MANET, WSN, WMN, VANET*. CRC press, 2016.

[59] Y.-H. Ho, C.-H. Lin, and L.-J. Chen, "On-demand misbehavior detection for vehicular ad hoc network," *International Journal of Distributed Sensor Networks*, vol. 12, no. 10, p. 1550147716673928, 2016.

[60] Y. Zhang, L. Lazos, and W. Kozma, "Amd: Audit-based misbehavior detection in wireless ad hoc networks," *IEEE transactions on mobile computing*, vol. 15, no. 8, pp. 1893–1907, 2012.

[61] M. Kadam and S. Limkar, "D&pmv: New approach for detection and prevention of misbehave/malicious vehicles from vanet," in *Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2013*. Springer, 2014, pp. 293–303.

[62] O. Abdel Wahab, H. Otrok, and A. Mourad, "A cooperative watchdog model based on dempster–shafer for detecting misbehaving vehicles," 2014.

[63] B. A. S. Al-Rimy, M. A. Maarof, M. Alazab, S. Z. M. Shaid, F. A. Ghaleb, A. Almalawi, A. M. Ali, and T. Al-Hadhrami, "Redundancy coefficient gradual up-weighting-based mutual information feature selection technique for crypto-ransomware early detection," *Future Generation Computer Systems*, vol. 115, pp. 641–658, 2021.

[64] A. Jaeger, N. Bißmeyer, H. Stübing, and S. A. Huss, "A novel framework for efficient mobility data verification in vehicular ad-hoc networks," *International Journal of Intelligent Transportation Systems Research*, vol. 10, no. 1, pp. 11–21, 2012.

[65] R. K. Schmidt, T. Leinmüller, E. Schoch, A. Held, and G. Schäfer, "Vehicle behavior analysis to enhance security in vanets," in *Proceedings of the 4th IEEE Vehicle-to-Vehicle Communications Workshop (V2VCOM2008)*. Citeseer, 2008.

[66] N. Bißmeyer, S. Mauthofer, K. M. Bayarou, and F. Kargl, "Assessment of node trustworthiness in vanets using data plausibility checks with particle filters," in *2012 IEEE Vehicular Networking Conference (VNC)*. IEEE, 2012, pp. 78–85.

[67] S. K. Dhurandher, M. S. Obaidat, A. Jaiswal, A. Tiwari, and A. Tyagi, "Vehicular security through reputation and plausibility checks," *IEEE Systems Journal*, vol. 8, no. 2, pp. 384–394, 2013.

[68] C. Velasco-Gallego and I. Lazakis, "Real-time data-driven missing data imputation for short-term sensor data of marine systems. a comparative study," *Ocean Engineering*, vol. 218, p. 108261, 2020.

[69] S. A. Almalki, A. Abdel-Rahim, and F. T. Sheldon, "Disrupting the cooperative nature of intelligent transportation systems," in *2022 IEEE World AI IoT Congress (AIIoT)*. IEEE, 2022, pp. 131–137.

[70] M. Sweet, "Does traffic congestion slow the economy?" *Journal of Planning Literature*, vol. 26, no. 4, pp. 391–404, 2011.

[71] B. M. Williams and A. Guin, "Traffic management center use of incident detection algorithms: Findings of a nationwide survey," *IEEE Transactions on intelligent transportation systems*, vol. 8, no. 2, pp. 351–358, 2007.

[72] F. A. Ghaleb, B. A. S. Al-Rimy, A. Almalawi, A. M. Ali, A. Zainal, M. A. Rassam, S. Z. M. Shaid, and M. A. Maarof, "Deep kalman neuro fuzzy-based adaptive broadcasting scheme for vehicular ad hoc network: A context-aware approach," *IEEE Access*, vol. 8, pp. 217 744–217 761, 2020.

[73] J. Firl, H. Stübing, S. A. Huss, and C. Stiller, "Marv-x: Applying maneuver assessment for reliable verification of car-to-x mobility data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1301–1312, 2013.

[74] J. Petit and S. E. Shladover, "Potential cyberattacks on automated vehicles," *IEEE Transactions on Intelligent transportation systems*, vol. 16, no. 2, pp. 546–556, 2014.

[75] R. W. van der Heijden and F. Kargl, "Open issues in differentiating misbehavior and anomalies for vanets," *Proceedings of 2nd GI/ITG KuVS Fachgespräch Inter-Vehicle Communication. Vehicular Lab, University of Luxembourg*, pp. 24–26, 2014.

[76] A. F. Santamaria, C. Sottile, F. De Rango, and M. Voznak, "Road safety alerting system with radar and gps cooperation in a vanet environment," in *Wireless Sensing, Localization, and Processing IX*, vol. 9103. SPIE, 2014, pp. 120–133.

[77] R. A. Uzcátegui, A. J. De Sucre, and G. Acosta-Marum, "Wave: A tutorial," *IEEE Communications magazine*, vol. 47, no. 5, pp. 126–133, 2009.

[78] J. Hou, J. Liu, L. Han, and J. Zhao, "Secure and efficient protocol for position-based routing in vanets," in *2012 IEEE International Conference on Intelligent Control, Automatic Detection and High-End Equipment*. IEEE, 2012, pp. 142–148.

[79] V. Milanés, S. E. Shladover, J. Spring, C. Nowakowski, H. Kawazoe, and M. Nakamura, "Cooperative adaptive cruise control in real traffic situations," *IEEE Transactions on intelligent transportation systems*, vol. 15, no. 1, pp. 296–305, 2013.

[80] S. Dietzel, J. Petit, G. Heijenk, and F. Kargl, "Graph-based metrics for insider attack detection in vanet multihop data dissemination protocols," *IEEE transactions on vehicular technology*, vol. 62, no. 4, pp. 1505–1518, 2012.

[81] K. Z. Ghafoor, J. Lloret, K. A. Bakar, A. S. Sadiq, and S. A. B. Mussa, "Beaconing approaches in vehicular ad hoc networks: A survey," *Wireless personal communications*, vol. 73, no. 3, pp. 885–912, 2013.

[82] K. Golestan, F. Sattar, F. Karray, M. Kamel, and S. Seifzadeh, "Localization in vehicular ad hoc networks using data fusion and v2v communication," *Computer Communications*, vol. 71, pp. 61–72, 2015.

[83] K. Liu, H. B. Lim, E. Frazzoli, H. Ji, and V. C. Lee, "Improving positioning accuracy using gps pseudorange measurements for cooperative vehicular localization," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 6, pp. 2544–2556, 2013.

[84] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proceedings of the IEEE*, vol. 97, no. 2, pp. 427–450, 2009.

[85] J. Zhang, "A survey on trust management for vanets," in *2011 IEEE International Conference on Advanced Information Networking and Applications*. IEEE, 2011, pp. 105–112.

[86] J. Grover, N. K. Prajapati, V. Laxmi, and M. S. Gaur, "Machine learning approach for multiple misbehavior detection in vanet," in *International conference on advances in computing and communications*. Springer, 2011, pp. 644–653.

[87] T. Leinmuller, E. Schoch, and F. Kargl, "Position verification approaches for vehicular ad hoc networks," *IEEE Wireless Communications*, vol. 13, no. 5, pp. 16–21, 2006.

[88] Y. Park and H. Kim, "Application-level frequency control of periodic safety messages in the ieee wave," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 4, pp. 1854–1862, 2012.

[89] R. G. Engoulou, M. Bellaïche, S. Pierre, and A. Quintero, "Vanet security surveys," *Computer Communications*, vol. 44, pp. 1–13, 2014.

[90] F. A. Ghaleb, A. Zainal, M. A. Rassam, and A. Abraham, "Improved vehicle positioning algorithm using enhanced innovation-based adaptive kalman filter," *Pervasive and Mobile Computing*, vol. 40, pp. 139–155, 2017.

[91] F. A. Ghaleb, A. Zainal, M. A. Rassam, and F. Saeed, "Driving-situation-aware adaptive broadcasting rate scheme for vehicular ad hoc network," *Journal of Intelligent & Fuzzy Systems*, vol. 35, no. 1, pp. 423–438, 2018.

[92] F. A. Ghaleb, A. Zainal, M. A. Maroof, M. A. Rassam, and F. Saeed, "Detecting bogus information attack in vehicular ad hoc network: a context-aware approach," *Procedia Computer Science*, vol. 163, pp. 180–189, 2019.

[93] N. Nikaein, S. K. Datta, I. Marecar, and C. Bonnet, "Application distribution model and related security attacks in vanet," in *International Conference on Graphic and Image Processing (Icgip 2012)*, vol. 8768. SPIE, 2013, pp. 37–42.

[94] Y.-M. Chen and Y.-C. Wei, "A beacon-based trust management system for enhancing user centric location privacy in vanets," *Journal of Communications and Networks*, vol. 15, no. 2, pp. 153–163, 2013.

[95] S. Ercan, M. Ayaida, and N. Messai, "Misbehavior detection for position falsification attacks in vanets using machine learning," *IEEE Access*, vol. 10, pp. 1893–1904, 2021.

[96] G. Brown, A. Pocock, M.-J. Zhao, and M. Luján, "Conditional likelihood maximisation: a unifying framework for information theoretic feature selection," *The journal of machine learning research*, vol. 13, no. 1, pp. 27–66, 2012.

[97] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," *ACM computing surveys (CSUR)*, vol. 50, no. 6, pp. 1–45, 2017.

[98] M. Bennasar, Y. Hicks, and R. Setchi, "Feature selection using joint mutual information maximisation," *Expert Systems with Applications*, vol. 42, no. 22, pp. 8520–8532, 2015.

[99] Z.-G. Chen, H.-S. Kang, S.-N. Yin, and S.-R. Kim, "Automatic ransomware detection and analysis based on dynamic api calls flow graph," in *Proceedings of the International Conference on Research in Adaptive and Convergent Systems*, 2017, pp. 196–201.

[100] B. A. S. Al-rimy, M. Kamat, F. A. Ghaleb, F. Rohani, S. A. Razak, M. A. Shah *et al.*, "A user mobility-aware fair channel assignment scheme for wireless mesh network," in *Computational Science and Technology*. Springer, 2020, pp. 531–541.

[101] Z. K. Maseer, R. Yusof, S. A. Mostafa, N. Bahaman, O. Musa, and B. A. S. Al-rimy, "Deepiot. ids: hybrid deep learning for enhancing iot network intrusion detection," *CMC-Computers Materials & Continua*, vol. 69, no. 3, pp. 3945–3966, 2021.

[102] S. A. Almalki, A. Abdel-Rahim, and F. T. Sheldon, "Adaptive ids for cooperative intelligent transportation systems using deep belief networks," *Algorithms*, vol. 15, no. 7, p. 251, 2022.

[103] M. Talal, K. N. Ramli, A. Zaidan, B. Zaidan, and F. Jumaa, "Review on car-following sensor based and data-generation mapping for safety and traffic management and road map toward its," *Vehicular Communications*, vol. 25, p. 100280, 2020.

[104] Y. A. Ahmed, S. Huda, B. A. S. Al-rimy, N. Alharbi, F. Saeed, F. A. Ghaleb, and I. M. Ali, "A weighted minimum redundancy maximum relevance technique for ransomware early detection in industrial iot," *Sustainability*, vol. 14, no. 3, p. 1231, 2022.

[105] U. Urooj, B. A. S. Al-rimy, A. Zainal, F. A. Ghaleb, and M. A. Rassam, "Ransomware detection using the dynamic analysis and machine learning: A survey and research directions," *Applied Sciences*, vol. 12, no. 1, p. 172, 2021.

[106] M. N. Olaimat, M. A. Maarof, and B. A. S. Al-rimy, "Ransomware anti-analysis and evasion techniques: A survey and research directions," in *2021 3rd international cyber resilience conference (CRC)*. IEEE, 2021, pp. 1–6.

[107] F. Azam, S. Kumar, and N. Priyadarshi, "Privacy and authentication schemes in vanets using blockchain: A review and a framework to mitigate security and privacy issues," *AI Enabled IoT for Electrification and Connected Transportation*, pp. 127–145, 2022.

[108] A. Alharthi, Q. Ni, and R. Jiang, "A privacy-preservation framework based on biometrics blockchain (bbc) to prevent attacks in vanet," *IEEE Access*, vol. 9, pp. 87 299–87 309, 2021.

[109] F. A. Ghaleb, F. Saeed, E. H. Alkhammash, N. S. Alghamdi, and B. A. S. Al-Rimy, "A fuzzy-based context-aware misbehavior detecting scheme for detecting rogue nodes in vehicular ad hoc network," *Sensors*, vol. 22, no. 7, p. 2810, 2022.

[110] M. A. Alsoufi, S. Razak, M. M. Siraj, I. Nafea, F. A. Ghaleb, F. Saeed, and M. Nasser, "Anomaly-based intrusion detection systems in iot using deep learning: A systematic literature review," *Applied Sciences*, vol. 11, no. 18, p. 8383, 2021.

[111] E. Qafzezi, K. Bylykbashi, P. Ampririt, M. Ikeda, K. Matsuo, and L. Barolli, "A fuzzy-based approach for resource management in sdn-vanets: Effect of trustworthiness on assessment of available edge computing resources," *Journal of High Speed Networks*, vol. 27, no. 1, pp. 33–44, 2021.

[112] S. Sultan, Q. Javaid, A. J. Malik, F. Al-Turjman, and M. Attique, "Collaborative-trust approach toward malicious node detection in vehicular ad hoc networks," *Environment, Development and Sustainability*, vol. 24, no. 6, pp. 7532–7550, 2022.

[113] Y. Alghofaili, A. Albattah, N. Alrajeh, M. A. Rassam, and B. A. S. Al-rimy, "Secure cloud infrastructure: A survey on issues, current solutions, and open challenges," *Applied Sciences*, vol. 11, no. 19, p. 9005, 2021.

[114] B. A. Khalaf, S. A. Mostafa, A. Mustapha, M. A. Mohammed, M. A. Mahmoud, B. A. S. Al-Rimy, S. Abd Razak, M. Elhoseny, and A. Marks, "An adaptive protection of flooding attacks model for complex network environments," *Security and Communication Networks*, vol. 2021, 2021.

[115] A. A. Cook, G. Mısırlı, and Z. Fan, "Anomaly detection for iot time-series data: A survey," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6481–6494, 2019.

[116] K. Akshaya and T. Sarath, "Detecting sybil node in intelligent transport system," in *Innovative Data Communication Technologies and Application*. Springer, 2022, pp. 595–607.

[117] A. Alsarhan, M. Alauthman, E. Alshdaifat, A.-R. Al-Ghuwairi, and A. Al-Dubai, "Machine learning-driven optimization for svm-based intrusion detection system in vehicular ad hoc networks," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–10, 2021.