ANALYSIS AND TECHNIQUES OF CYBERATTACK TYPES CLASSIFICATION IN SMART GRIDS

by

Victor Odion Ijeh

A thesis submitted to the
School of Graduate and Postdoctoral Studies in partial
fulfillment of the requirements for the degree of

Master of Applied Science in Electrical and Computer Engineering

The Faculty of Engineering and Applied Science

University of Ontario Institute of Technology (Ontario Tech University)

Oshawa, Ontario, Canada

September 2023

THESIS EXAMINATION INFORMATION

Submitted by: **Victor Odion Ijeh**

Master of Applied Science in Electrical and Computer Engineering

Thesis title: ANALYSIS AND TECHNIQUES OF CYBERATTACK TYPES CLASSIFICATION IN SMART GRIDS

An oral defense of this thesis took place on September 15, 2023 in front of the following examining committee:

Examining Committee:

Chair of Examining Committee	Dr. Ghaus Rizvi
Research Supervisor	Dr. Walid Morsi Ibrahim
Examining Committee Member	Dr. Mohamed Youssef
Thesis Examiner	Dr. Meaghan Charest-Finn, Ontario Tech University

The above committee determined that the thesis is acceptable in form and content and that a satisfactory knowledge of the field covered by the thesis was demonstrated by the candidate during an oral examination. A signed copy of the Certificate of Approval is available from the School of Graduate and Postdoctoral Studies.

ABSTRACT

The smart electric grids rely on integrating the information and communication technologies (ICT) into the electric power grid infrastructure to facilitate the exchange of information for an enhanced and economic operation. Such integration of ICT into the existing electric grids makes them vulnerable to cybersecurity threats, ranging from data breaches to service disruptions. The work in this thesis investigates the use of machine learning techniques to detect and classify such cyberattacks. A novel approach that uses a fine tree bagging ensemble learning technique to detect and classify the cyberattack types from normal and power quality disturbances is developed. The proposed approach extracts the relevant features for classifying different cyber-attack types such as message suppression, denial-of-service and data manipulation. The proposed approach is tested on a publicly available dataset and the results are compared to three other machine learning techniques, namely decision tree, nearest neighbor, and support vector machine. The results have shown that the proposed approach is very effective in the detection and the classification of the cyberattack types as well as it is insensitive to the selection of the training and the testing datasets.

Keywords: Classification; cyberattack; data manipulation; ensemble learning; substation automation.

AUTHOR'S DECLARATION

I hereby declare that this thesis consists of original work of which I have authored. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize the University of Ontario Institute of Technology (Ontario Tech University) to lend this thesis to other institutions or individuals for the purpose of scholarly research. I further authorize University of Ontario Institute of Technology (Ontario Tech University) to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research. I understand that my thesis will be made electronically available to the public.

Victor Odion Ijeh

STATEMENT OF CONTRIBUTIONS

The main contribution of this thesis is to introduce a novel method for smart grid cyberattack classification that not only detects different cyberattacks but it can also identify the type of cyberattack.

In this research, Predictor Importance (PI) technique is calculated and is used to identify the key relevant physical and network features needed to classify the cyberattack types.

A novel approach is introduced that utilizes a Fine Tree Bagging-based Ensemble learning approach with hyperparameter optimization of parameters such as k-fold and number of learners discussed in detail in Chapter#5.

This study showcases the efficacy of the proposed methodology by computing various evaluation criteria and comparing it with other machine learning methodologies as presented in Chapter#6.

The proposed approach is insensitive to the selection of the training and the testing datasets and hence it overcomes the limitations of the existing approaches that suffer overfitting.

ACKNOWLEDGEMENTS

Firstly, I would like to thank God Almighty for his grace, blessing, protection, and sustenance throughout my life and being my guide in completing my thesis.

I would like to specially express my sincere gratitude to my supervisor Prof. Walid Morsi Ibrahim for giving me the opportunity and providing me with all the necessary resources. His intellectual guidance, assistance, supervision, and advice made it possible for me to complete my thesis.

I would also like to appreciate all my colleagues at the Smart Grid and Electric Vehicle Research Lab at Ontario Tech University: Kripa Mary Jose, Ahmad Abu Nassar and Matthew Oinonen for their immeasurable assistance during my studies.

Last but not the least, I would like to wholeheartedly express my deepest appreciation to my parents for their unwavering love, steadfast support, uplifting prayers and constant encouragement throughout my life. Special thanks to my brothers, Lucky and Henry for their calls, funny messages, care and support.

TABLE OF CONTENTS

THESIS EXAMINATION INFORMATION ..... **ii**
ABSTRACT ..... **iii**
AUTHOR'S DECLARATION ..... **iv**
STATEMENT OF CONTRIBUTIONS ..... **v**
ACKNOWLEDGEMENTS ..... **vi**
TABLE OF CONTENTS ..... **vii**
List of Tables ..... **xi**
List of Figures ..... **xiii**
Abbreviations ..... **xvii**
1. Introduction 1
1.1. Background 1
1.2. Trends of Cyberattack Worldwide 3
1.3. Vulnerable Assets in Smart Grid Infrastructure 6
1.4. Impacts of Cyberattack on Smart Grid 8
1.5. Problem Statement and Motivation 9
1.6. Contributions 11
1.7. Thesis Organization 11
2. Literature Review 14
2.1. Introduction 14
2.2. Previous Work on Detection of Cyberattacks. 14
2.2.1. Non-Machine Learning Based Approach 15
2.2.2. Machine Learning Based Approach 16
2.3. Previous Work on Detection and Classification of Cyberattacks 20
2.3.1. Non-Machine Learning Based Approach 20
2.3.2. Machine Learning Based Approach 22
2.4. Research Gaps 23
2.5. Summary 25
3. Machine Learning Methodologies used in Smart Grid 26
3.1 Introduction 26
3.2. Machine Learning-Based Algorithms 26
3.3. Supervised learning Methods 27

3.4. Decision Tree Algorithm 27
3.4.1. Decision Tree Attribute Selection Measure 28
3.4.2. Strengths and Limitations of Decision Trees 29
3.5. Support Vector Machine Algorithm 30
3.5.1. Hyperplane and Support Vectors in the SVM algorithm 30
3.5.2. Strengths and Limitations of Support Vector Machine Algorithms 31
3.6. K-Nearest Neighbor Algorithm 32
3.6.1. Selecting K in the KNN Algorithm 33
3.6.2. Strength and Limitation of KNN Algorithms 34
3.7. Summary 35
4. Analysis of the IEC 61850 Substation Communication Standard 36
4.1. Introduction to IEC 61850 Standard 36
4.2. Communication Architecture of IEC 61850 38
4.3. IEC 61850 Information Model 39
4.3.1. Logical device 40
4.3.2. Logical Nodes 40
4.3.3. Data Class 42
4.3.4. Data Attribute 42
4.3.5. Naming Convention in IEC 61850 43
4.4. Substation Configuration Language 44
4.5. Overview and Configuration of GOOSE and Sampled Values in IEC 61850 45
4.5.1. IEC 61850 Publisher-Subscriber Architecture 46
4.5.2. Retransmission Mechanism 47
4.5.3. Generic Object-Oriented Substation Events (GOOSE) 48
4.5.2. Operation of the GOOSE protocol 48
4.5.3. GOOSE frame 49
4.5.4. Sampled Values (SV) 50
4.6. Wireshark 50
5. Model for the Detection and Classification of Cyberattacks in IEC 61850 Substation Automation Systems 53
5.1. Introduction 53
5.2. Attack Types Description 53
5.2.1. FDIA Attack 54

5.2.2. Data Manipulation Attack 57
5.2.3. Message Suppression Attacks 59
5.2.4. Denial-of-Service Attacks 60
5.2.5. Replay Attacks 62
5.3. Feature description 62
5.3.1. Network Features 62
5.3.2. Physical Features 65
5.4. Data Collection 66
5.5. Data Preprocessing 67
5.6. Fine Tree-based Bagging Ensemble (FTBE) Approach for Cyberattack Classification 76
5.6.1. Fine Tree Model 77
5.6.2. Finding best split 78
5.6.3. K-fold Cross Validation 79
5.6.4. Bagging-based Ensemble Classifier 81
5.6.5. Fine Tree Bagging-Based Ensemble (FTBE) Approach 83
5.6.6. Number of Learners 84
5.6.7. Predictor Importance (PI) 86
5.7. Evaluation Metrics 87
5.7.1. Accuracy Measure 87
5.7.2. Precision 87
5.7.3. Recall 88
5.7.4. F1-score 88
5.7.5. Confusion Matrix 88
5.8. Summary 89
6. Results and Evaluation 91
6.1. Introduction 91
6.2. Test System Description 91
6.2.1. Test System 1 92
6.2.2. Test System 2 94
6.3. Scenario description of the Test Systems dataset 96
6.4. Training and Testing of Automated Classification 109
6.5. Results of Implementing the Proposed Approach on the 4-IED Dataset 110

6.5.1. Confusion Matrices and Features Identification for Cyberattack Types .. 111
6.5.2. Sensitivity and Error Analysis 115
6.6. Result of Implementing the Proposed Approach on the 18-IED Dataset..... 117
6.6.1. Confusion Matrices and Features Identification for Cyberattack Types .. 119
6.6.2. Sensitivity and Error Analysis 122
6.7. Summary..... 124
7. Conclusion and Recommendations 125
7.1. Conclusion..... 125
7.2. Recommendations 127
7.3. Future Work 127
References..... 128

List of Tables

Chapter 1

Table 1.1: Cyberattacks and their Impacts to some Smart grids around the world. 10

Chapter 2

Table 2.1: Literature Review Table of Detection and Classification of Cyberattacks in Smart Grids 24

Chapter 4

Table 4.1: Scope and Outline of the IEC 61850 standard..... 37

Table 4.2: Categorisation of Logical Nodes in IEC 61850..... 41

Table 4.3: Logical Node Classes in IEC 61850 substation..... 41

Table 4.4: Data Objects in IEC 61850 42

Chapter 5

Table 5.1: 4-IED Dataset Network Features Description 72

Table 5.2: 4-IED Dataset Physical Features Description..... 72

Table 5.3:Network Features Description of the 18-IED Dataset 74

Table 5.4: Physical Features Description of the 18-IED Dataset 75

Table 5.5: Confusion matrix 89

Chapter 6

Table 6.1: Scenario description of the 4-IED dataset 99

Table 6.2: Scenario description of the 18 IED dataset..... 108

Table 6.3: 18-IED Dataset Combinations for Training and Testing..... 110

Table 6.4: Comparative analysis of the Accuracy Results of the 4-IED Dataset 111

Table 6.5: Percentage Change when Number of Learners is Constant with Kfold 2 to 12 116

Table 6.6: Percentage Change when KFOLD is Constant with Number of Learners 2 to 10 116

Table 6.7: Comparative Analysis of the Accuracy Results of the 18-IED Dataset 118

List of Figures

Chapter 1

Figure 1.1: National Institute of Standards and Technology (NIST) SG Model 2

Figure 1.2: Overview of cyberattacks on Smart Grids since 2010. 4

Figure 1.3: Cyber Incidents on Critical Infrastructure Reported to the DHS, Industrial Control Systems Cyber Emergency Response Team (ICS-CERT) 8

Chapter 2

Figure 2.1: Architecture of the Anomaly Detection System through Normal Behavior Profiling 18

Chapter 3

Figure 3.1: Decision Tree 28

Figure 3.2: Representation of SVM in a 2-dimensional space 31

Figure 3.3: Implementation of the Algorithm for K=5 (a) Before KNN and (b) After KNN 33

Chapter 4

Figure 4.1: Architecture of an IEC 61850 based substation 38

Figure 4.2: IEC 61850 device representation..... 40

Figure 4.3: Default Naming scheme of IEC 61850 43

Figure 4.4: Information flow of the configuration process..... 45

Figure 4.5: Publisher and subscriber communication 46

Figure 4.6: Retransmission mechanism in publisher-subscriber architecture 47

Figure 4.7: Goose message format 49

Figure 4.8: Wireshark user interface of the capture of a GOOSE packet. 51

Chapter 5

Figure 5.1: FDIA in smart Grid 55

Figure 5.2: A taxonomy of FDIA attacks against various power system control and operation blocks 56

Figure 5.3: Network Diagram for Data Manipulation Malware Attack 58

Figure 5.4: Test bed for DoS Attack 61

Figure 5.5: Summary of the Physical Features of the Primary Plant, Categorized by their Horizontal Levels. 70

Figure 5.6: Process of the K-fold cross validation method 81

Figure 5.7: High Bias and High Variance Performance of a Weak Learner 82

Figure 5.8: Plot Performance of an Ensemble Learner 83

Figure 5.9: Feature Extraction and Class Prediction of the Proposed Approach 85

Chapter 6

Figure 6.1: 66/22kV Substation Test System used to Generate the 4-IED Dataset. 92

Figure 6.2: Structure of Communication Networks in the Test system 93

Figure 6.3: Single-line diagram of the 66/11kV substation automation system 94

Figure 6.4: Communication Framework for generating attack-free and attack induced GOOSE traces. 95

Figure 6.5: Wireshark capture of replay attack GOOSE packets from IED 1. 97

Figure 6.6: Wireshark capture of FDIA GOOSE packets from IED 1. 97

Figure 6.7: Wireshark capture of replay attack GOOSE packets from IED 2. 98

Figure 6.8: Wireshark capture of FDIA GOOSE packets from IED 2. 98

Figure 6.9: LIED10 injects a GOOSE frame (No. 588) with a value of 380 for phase A current magnitude at 11.9 seconds. 101
Figure 6.10: LIED10 injects a GOOSE frame (No. 1175) with a value of 270 for phase B current magnitude at 22.5 seconds. 101
Figure 6.11: LIED10 injects a GOOSE frame (No. 1771) with a value of 360 for phase C current magnitude at 33.1 seconds. 102
Figure 6.12: LIED11 injects malicious GOOSE frame (No. 597) changing the circuit breaker status from FALSE to TRUE ('tripped') at 12.3 seconds. 102
Figure 6.13: LIED11 replays valid GOOSE frames (No. 2734 and No. 2764) with "open" (True) circuit breaker data at times 53.8 sec and 54.8 sec. 103
Figure 6.14: LIED22 replays fault current measurements (No. 7847, No. 7901, No. 7955, No. 8009, and No. 8063) between times 155.8 sec and 159.88 sec. 103
Figure 6.15: Denial-of-Service (DoS) attack on LIED10 104
Figure 6.16: LIED10 injects a GOOSE frame (No. 542) with Stnum=9999 and Sqnum=10 at 13.9 secs. 104
Figure 6.17: LIED10 injects a GOOSE frame (No. 784) with Stnum=5 and Sqnum=15 at 18.9 secs. 105
Figure 6.18: LIED10 replays a GOOSE frame (No. 534) with Stnum=9999 and Sqnum=0 at 10.4 secs. 105
Figure 6.19: LIED12 replays a GOOSE frame (No. 774) with Stnum=5 and Sqnum=0 at 15.5 secs. 106
Figure 6.20: LIED10 injects a GOOSE frame (No. 534) with Stnum=9999 and Sqnum=0 at 10.4 secs. 106

Figure 6.21: LIED12 injects a GOOSE frame (No. 774) with Stnum=5 and Sqnum=0 at 15.5 secs. 106

Figure 6.22: LIED10 injects a GOOSE frame (No. 556) with Sqnum=9999 at time= 12.7 sec. 107

Figure 6.23: LIED11 injects a GOOSE frame (No. 542) with Stnum=9999 and Sqnum=0 at 11.3 sec. 107

Figure 6.24: LIED11 modifies the CB-11 Boolean value from '1' to '0' and injects the modified GOOSE frame (No. 792) at 16.3 secs. 108

Figure 6.25: Classification results for case 1. (a) Confusion Matrix (b) predictor importance. 113

Figure 6.26: Classification results for case 2. (a) Confusion Matrix (b) predictor importance. 114

Figure 6.27: Classification results of case 1. (a) Confusion Matrix and (b) predictor importance. 120

Figure 6.28: Classification results of case 2. (a) Confusion Matrix and (b) predictor importance. 120

Figure 6.29: Classification results of case 3. (a) Confusion Matrix and (b) predictor importance. 121

Figure 6.30: Classification results of case 4. (a) Confusion Matrix and (b) predictor importance. 121

Figure 6.31: Classification accuracies for cases 1 to 4. (a) different number of learners and (b) different k-folds. 123

Abbreviations

AGC	Automatic Generation Control
allData	All Data Types
AMI	Advanced Metering Infrastructure
APPID	Application Identifier
ASN.1	Abstract Syntax Notation One
AVR	Automatic Voltage control
BCV	Boolean Control Signal Values
CART	Classification and Regression Tree
CDC	Common Data Classes
CFI	Canonical Format Indicator
CSP	Candidate Split Position
CSV	Comma Separated Values
CT	Current Transformer
DatSet	Data Set
DM	Data Manipulation
DMS	Distribution Management System
DNP	Distributed Network Protocol
DoS	Denial of Service
EMS	Energy Management System
FACTS	Flexible Alternating Current Transmission System
FDIA	False Data Injection Attack
FTBE	Fine Tree Bagging Ensemble

GOOSE	Generic Object-Oriented Substation Event
GoosePDU	Generic Object-Oriented Substation Event Protocol Data Unit
GPS	Global Positioning System
HMI	Human Machine Interface
ICS	Industrial Control System
ICT	Information and Communication Technology
IEC	International Electronic Commission
IED	Intelligent Electronic Devices
IEEE	Institute od Electrical and Electronics Engineers
IoT	Internet of Things
KF	Kalman Filter
KNN	K-Nearest Neighbor
LIED	Line Feeder Intelligent Electronic Devices
LN	Logical Nodes
MAC dest	Destination Media Access Control
MAC src	Source Media Access Control
MS	Message Suppression
MTU	Master Terminal Unit
PCAP	Packet Capture
PCP	Priority Code Point
PDC	Phasor Data Concentrator
PI	Predictor Importance
PLC	Programmable Logic Circuit

PMU	Phasor Measurement Unit
RTU	Remote Terminal Unit
SAS	Substation Automation System
SCADA	Supervisory Control and Data Acquisition
SCL	Substation Configuration Language
Sqnum	Sequence number
Stnum	Status number
SV	Sampled Values
SVM	Support Vector Machine
TPID	Tag Protocol Identifier
UDP	User Datagram Protocol
USB	Universal Serial Bus
VID	Virtual Local Area Network Identifier
VLAN	Virtual Local Area Network
VM	Virtual Machine
VT	Voltage Transformer
WLS	Weighted Least Squares
WSN	Wireless Sensor Network
XML	Extensible Markup Language

1. Introduction

1.1. Background

Smart grid (SG) is a modernized version of the legacy electric grid. The smart grid integrates the information and communication technology (ICT) into the electric power grid for an efficient and an economic operation. This integration involves sharing information, between intelligent electronic devices (IEDs) and the supervisory control and data acquisition (SCADA) systems used in the substation automation systems. By exchanging this information, the smart grid becomes more efficient, allows for smooth integration of renewable energy resources and ensures the cost-effective operation of its assets [1].

The model depicted in Figure 1.1 illustrates the structure of a smart grid. These grids possess unique characteristics when compared to the conventional grids, such as the ability for power to flow in both directions in real time and seamless communication between utility companies and consumers. Moreover, there is also a communication established between the distribution substations and the customers, which is made possible by devices, like IEDs, SCADA and switchgears [2]. This communication framework offers a multitude of advantages including automated metering, redundancy, maintenance capabilities, self-heal mechanisms, efficient energy management as well as improved reliability and security [3].

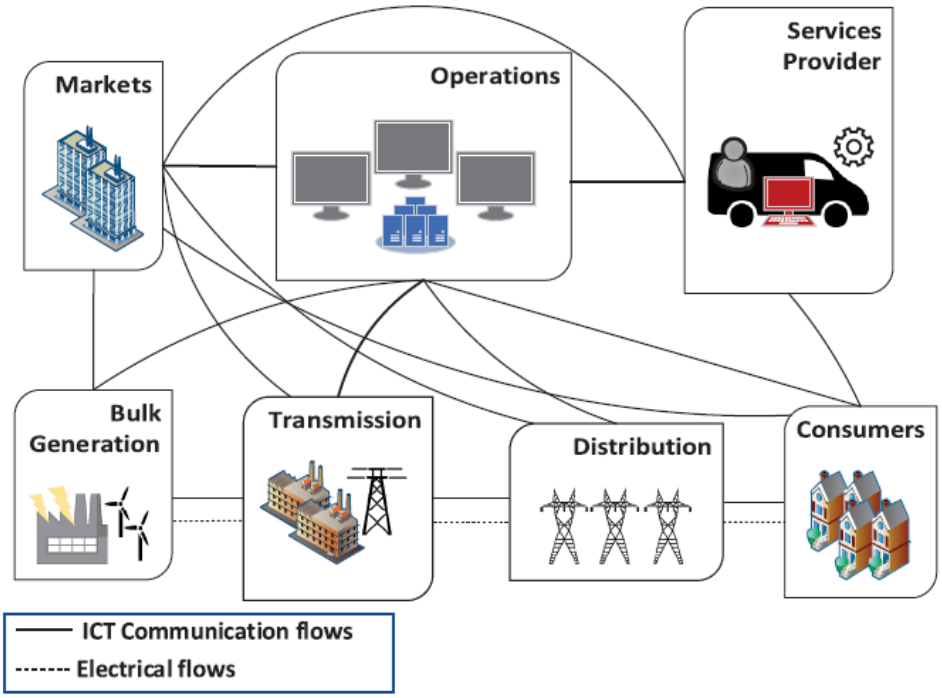


Figure 1.1: National Institute of Standards and Technology (NIST) SG Model [4]

With the advancements in ICT, the field of cybersecurity faces new challenges particularly when it comes to securing the electricity grid infrastructure. The integration of Internet of Things (IoT) applications, industrial devices and Wireless Sensor Networks (WSNs) has exposed the electric power grid to cyber threats that can jeopardize the national security [5]. Unfortunately, these devices often lack built-in security measures against attacks leaving them vulnerable to breaches. Additionally, concerns arise from the utilization of devices such as smart meters that communicate autonomously without human involvement. Furthermore, the legacy systems such as the conventional SCADA systems may not have up to date security solutions in place.

1.2. Trends of Cyberattack Worldwide

The security of the control systems that manage the critical infrastructures has become a primary focus for cyber terrorism and warfare. Figure 1.2 shows a timeline summarizing major smart grid attacks around the world from 2010 to 2023. One of the notable cyberattacks occurred in 2010, when a malware referred to as Stuxnet targeted an Iranian nuclear enrichment centrifuges causing significant damage to their equipment [6]. The attack involved exploiting vulnerabilities on the substation computer system through a Universal Serial Bus (USB) drive and injecting malicious software into Siemens Programmable Logic Circuits (PLCs). This caused the centrifuges to spin at frequencies than usual leading to increased wear and tear. Additionally, the malware manipulated sensor readings to hide the attack from the operators.

In December 2015, Ukraine experienced a cyber-attack through the injection of a malware called BlackEnergy, which specifically targeted two western oblast power grids. As a result, 30 substations were disconnected for approximately three hours. This malicious act led to a power outage that impacted around 230,000 residents. Consequently individuals faced difficulties in reaching out to their utility providers as the attack disrupted phone communication, with power companies [7].

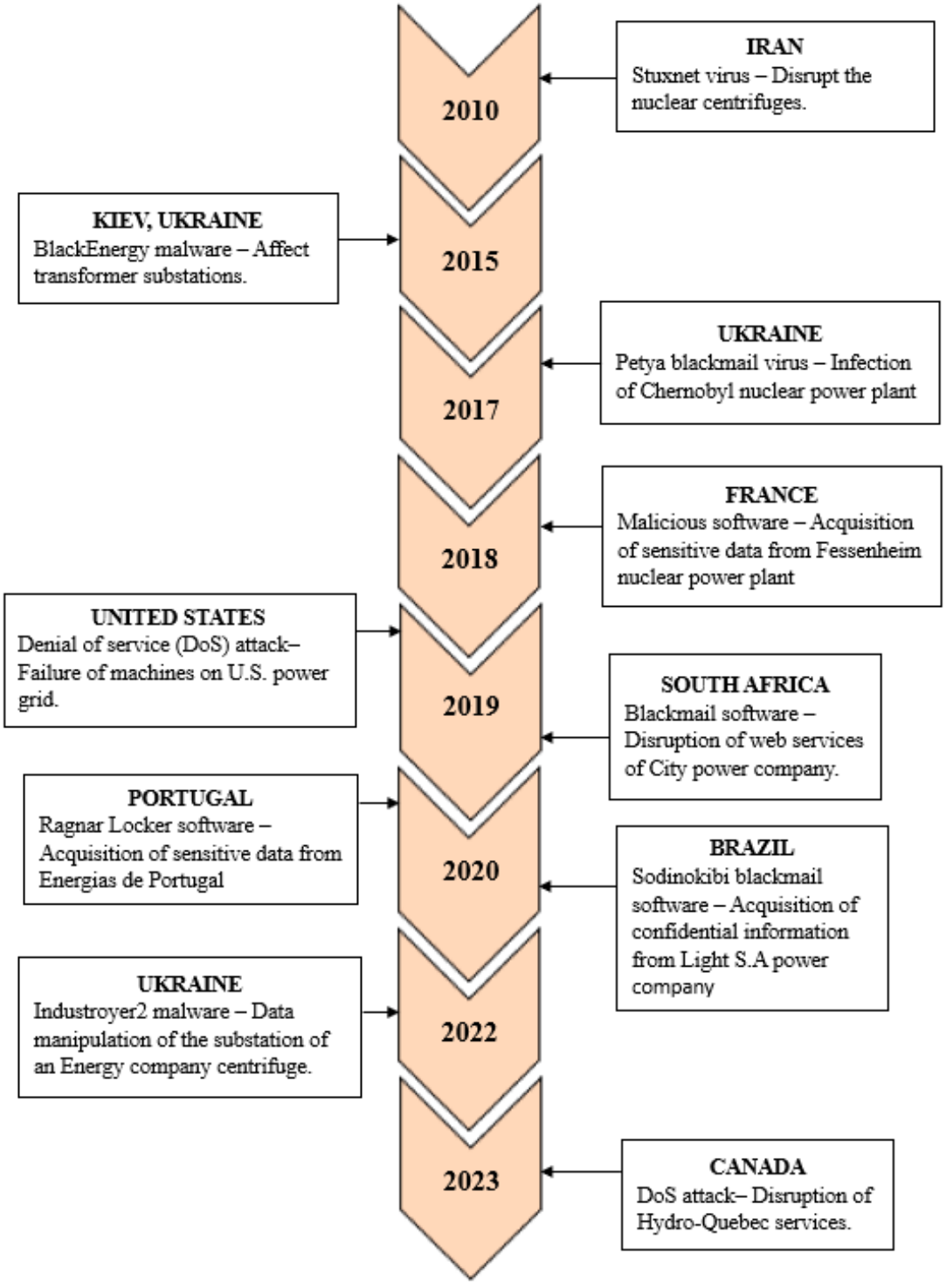


Figure 1.2: Overview of cyberattacks on Smart Grids since 2010 [8].

During the period between 2019 and 2020, a number of cyber attacks occurred. In March 2019, a power grid located in the western region of the United States experienced a denial of service (DoS) attack. The attack has affected a few industrial machines, which encountered failures lasting less than five minutes [9]. Another significant incident took place in Ukraine around April 2022. This particular attack involved a malware known as "Industroyer2", which was specifically designed to manipulate system commands. The malware directly targeted the utility equipment of a Ukrainian energy firm and sent commands to the substation devices responsible for regulating the electricity flow. Fortunately, the attack was discovered in time to prevent a power outage that could have affected approximately two million people [10].

In April 2023 Hydro Quebec, the electricity supplier in Quebec, Canada encountered a cyber incident that led to disruptions in their utility services while addressing power outages [11]. A hacking group with alleged ties to Russia claimed responsibility for the cyberattack, on this government owned power provider.

The number of cyberattack attempts is increasing in all sectors worldwide. The biggest increase however has been experienced in the manufacturing and utility sector, where the number of data breaches has constantly been on the rise [12]. From the cases of cyberattacks on smart grids mentioned above, the primary reason behind these losses is the vulnerability of the smart grid and the exploitation of these vulnerabilities through cyberattacks. When a smart grid is poorly designed to have countermeasures against cyberattacks, the integration of ICT and the increased application of IoT pose cybersecurity threats to such critical electricity infrastructure [8].

1.3. Vulnerable Assets in Smart Grid Infrastructure

Smart grids are complex systems that bring together physical networks, information technology (IT) and operational technology (OT) making them crucial infrastructures. Any weakness, whether internal or within the interconnected systems has the potential to jeopardize the grid security leading to power outages, financial losses and other significant consequences [13]. This integration encompasses systems such as Advanced Metering Infrastructure (AMI), Supervisory Control and Data Acquisition (SCADA) substations, synchrophasor systems, energy management systems (EMS), distribution management systems (DMS) and electric vehicle charging stations.

The Advanced Metering Infrastructure (AMI) plays a role in facilitating the bidirectional data exchange between end users and the utility companies [14]. It consists of three components; smart meters for monitoring power consumption, data collectors that store data from smart meters in specific geographic regions and the AMI headend, which acts as a central server where the utility companies aggregate and manage the collected data. The SCADA systems are primarily used for monitoring and controlling the automated functions. They include measuring instruments, logic controllers, a Master Terminal Unit (MTU), a communication network and a Human Machine Interface (HMI). The logic controllers work alongside sensors to manage the data flow efficiently by detecting anomalies and regulating the system components. These controllers communicate with the MTU using industrial protocols, like IEC 61850 [15]. The Human Machine Interface (HMI) plays a role in facilitating this interaction. Substations are a part of the electrical grid as they handle the transmission and distribution of power. They consist of devices such as Intelligent Electronic Devices (IEDs), Remote Terminal Units (RTUs), HMIs and

Global Positioning System (GPS) [14]. Synchrophasor systems are technologies utilized in modern grids. They incorporate Phasor Measurement Units (PMUs), Phasor Data Concentrators (PDCs) and a communication network. PMUs measure waveforms, while PDCs consolidate this data using standards like IEEE C37.118.2 and IEC 61850 for communication [16]. The Energy Management System (EMS) enables communication between utilities and third-party service providers allowing users to regulate their electricity usage. Finally, the Distribution Management Systems (DMS), which analyze real time electric distribution data to optimize power flows, prevent overloads and enhance outage management [17]. However, since DMS is integrated with IT infrastructure, it can be susceptible to cyber threats due to weaknesses in authentication, encryption and security measures.

Among these technologies, the Advanced Metering Infrastructure (AMI) and Supervisory Control and Data Acquisition (SCADA) systems are particularly vulnerable to cyberattacks [18]. The AMIs vulnerability stems from its consumer end devices and protocols that often lack security features like authentication, encryption and any excessive overhead due to the ease of use [19]. The SCADAs susceptibility lies in internal threats where an individual with system access can introduce malware similar to the Stuxnet incident in Iran [20]. Furthermore, the substations and the synchrophasor systems, which play a role in the functioning of the power grid are highly sought after by cyber attackers, as the communication protocols utilized in the systems such as the IEC 61850 are very vulnerable to attacks because of the deficiency of the protection scheme [21].

1.4. Impacts of Cyberattack on Smart Grid

The smart grid relies heavily on computer networks and other related technologies, which makes it susceptible to cyberattacks that can disrupt its operation. Internet-connected sensors, devices and networks are often targets of probing, espionage, ransomware attacks, theft and even physical destruction. Given the number of online nodes spread across wide geographical regions, the smart grid is highly exposed to significant cyber threats. Additionally, the power generation, transmission and usage are connected to other aspects of the economy such as manufacturing, transportation, healthcare and more. An attack on the power grid could result in disruptions to everyday production and livelihoods.

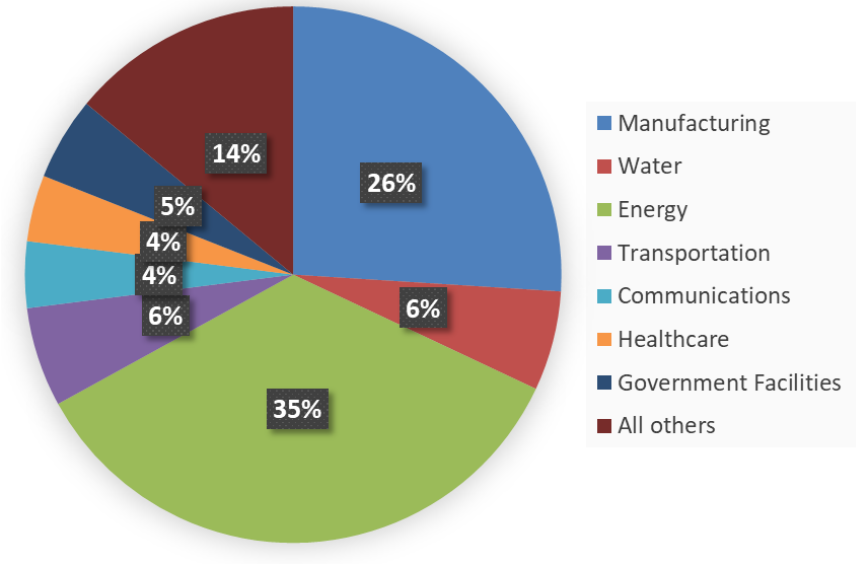


Figure 1.3: Cyber Incidents on Critical Infrastructure Reported to the DHS, Industrial Control Systems Cyber Emergency Response Team (ICS-CERT) [22]

As reported in [22], there has been an increase in cyberattacks on critical infrastructure within the USA. The energy industry has become a prime target accounting for 35% of these attacks as shown in Figure 1.3. These cyberattacks that target power grids can have severe impacts and could cost economies such as that of the USA $243 billion to

$1 trillion [23]. Successful cyberattacks can cause failures, synchronization loss, power outages, financial and social damages, data breaches, cascading failures and even complete blackouts [24]. In July 2019 Manhattan in New York experienced a blackout, which impacted other critical infrastructures [25]. The impact of such blackouts can result to loss of production, business closures, food spoilage, damage to electrical and electronic devices and the inability to operate certain systems in hospitals and other critical areas. It is worth noting that blackouts can also lead to property loss due to incidents, like arson and looting as observed during other occurrences [26].

1.5. Problem Statement and Motivation

Today's smart grid employ conventional intrusion detection and prevention systems (IDPS), which utilizes signature-based and anomaly methods to identify cyberattacks. However, recent reports indicate that these approaches are insufficient in safeguarding the grid [27]. Many reports have also highlighted the growing occurrence of cyberattack incidents and their reaching global effects. Table 1.1 provides a summary of major smart grid attacks and the resulting outcomes since 2010.

Table 1.1: Cyberattacks and their Impacts to some Smart grids around the world.

Year	Type of cyberattack	Country	Targeted assets	Impact and Consequence	Ref #
2010	Data Manipulation Attack	Iran	SCADA/ICS	Centrifuges for uranium enrichment rendered ineffective.	[6]
2014	Social engineering attack	South Korea	Communication network	5,986 phishing emails with malicious codes were sent to 3,571 nuclear plant employees.	[28]
2015	Distributed Denial of Service (DDoS) attack	Ukraine	Transformer substations	1.4 million people lost power, causing communication problems with power companies.	[7]
2016	FDIA	Israel	National power supply system	Israeli power facilities halt due to technical issues.	[29]
2017	Encrypting ransomware attack	Ukraine	Power plant computer systems	Abnormal operations occurred at multiple national power facilities due to the infection.	[30]
2020	Denial of Service attack	Italy	Internal IT network	IT network blockage caused customer service interruption.	[31]
2020	Data Manipulation Attack	Pakistan	AMI and Energy Management System	Data theft caused service interruptions.	[32]
2022	Command manipulation attack	Ukraine	Substation equipment	The attack-controlled power flow through direct interaction with utility equipment	[33]
2023	Denial of service attack	Canada	Outage management systems	Hydro-Québec's outage-checking platforms went down.	[11]

With the rise of the cyberattacks in smart grids, it becomes crucial to be able to categorize these attacks, for better understanding. This classification is vital as it enables an implementation of countermeasures to protect against current and future attack types. A timely identification of attack categories also facilitates responses and the implementation

of measures to prevent catastrophic incidents from spiraling out of control. Safeguarding the integrity of grid systems heavily relies on these countermeasures.

1.6. Contributions

The following summarizes the main contributions of this thesis:

- To develop an approach that uses machine learning techniques to detect the cyberattacks as well as provide classification of the attack types.
- To investigate and identify the key relevant physical and network features that is needed to classify the cyberattack types.
- To develop an approach that is effective in the detection and the classification of the cyberattacks without suffering the problem of data overfitting as in the existing approaches.
- To develop an approach that is insensitive to the selection of the training and the testing datasets.

1.7. Thesis Organization

This thesis includes seven chapters. Chapter 1 explains the trend and impact of cyberattacks on smart grid environment. Further, it explains the vulnerabilities of assets in the smart grid infrastructure and the importance of detecting and classifying cyberattacks followed by the problem statement and motivation. In conclusion, this research highlights the contributions made by the thesis.

Chapter 2 surveys different methodologies for the detection and classification of cyberattack types that were previously published in the literature. The advantages and

disadvantages of each method is presented and discussed. Finally, the main research gaps of the problem are outlined. Furthermore, the research direction to be pursued in this thesis is emphasized.

Chapter 3 is dedicated to the common Machine Learning-based methodologies utilized in cyberattack detection. This chapter explains the mathematical background of the three most popular machine learning techniques – Decision Tree, Support Vector Machine and K-Nearest neighbor. Finally, the benefits and drawbacks of each algorithm was also highlighted.

Chapter 4 discusses the IEC 61850 standard and its role in substation automation systems (SAS). The information model and the communication framework outlined by the standard is discussed. Additionally, the establishment of the communication within the devices in the substation and the structure and transmission of the GOOSE and Sampled value (SV) are also highlighted.

Chapter 5 describes the proposed model to detect and classify the cyberattacks in IEC 61850 SAS. Firstly, the data collection and the steps taken to preprocess it are discussed. Furthermore, the Fine Tree Bagging based Ensemble (FTBE) methodology for training and classifying attacks is discussed in detail. Lastly the process of the proposed approach and the metrics employed in this evaluation process are illustrated.

Chapter 6 provides the analysis of the approach put forward and showcases the findings. The algorithm is implemented, and tests were conducted using different selection of training and testing data. The results of different k-folds and the number of learners are presented and are discussed by comparing the classification accuracies, F-score, precision,

recall with the other machine learning methodologies highlighted in Chapter 3. Furthermore, the sensitivity of the proposed approach to the variation of the k-fold values and number of learners was explored.

Finally, Chapter 7 presents the main conclusion and recommendation regarding the classification of cyberattacks in IEC 61850 substation automation systems. It also presents potential avenues for future research and development that can build upon the findings and contributions of this study.

2. Literature Review

2.1. Introduction

In this chapter, previous work in the literature addressing the detection and the classification of the cyberattacks in smart grids are presented and are reviewed. The literature is divided into two categories; research focused solely on detecting the cyber attacks and research that includes both the detection and the classification of the cyberattacks. The first section of this chapter examines the previous work that used machine learning and non-machine learning methods to detect the cyber attacks. In the subsequent section, an overview of the previous work in the literature that proposed techniques that utilized machine learning based and non-machine-based approaches, for both detecting and classifying cyber attacks.

The main objective of the work presented in this thesis is to identify and classify the cyberattacks on smart grids. The literature review sheds light onto the previous efforts to classify cyberattack types through the use of machine learning. The intention is to evaluate these contributions, compare the effectiveness of specific techniques, and identify any limitations. Finally, this chapter highlights the research areas that require further exploration and sets the stage for the main findings of the research presented in this thesis.

2.2. Previous Work on Detection of Cyberattacks.

This section presents the existing methods that are introduced in the literature for the detection of cyberattacks. The previous work is classified into two different types: non-machine learning based approach and machine learning based approach as discussed below.

2.2.1. Non-Machine Learning Based Approach

Non-machine learning based approaches such as model-based algorithms are methods used in detecting cyberattacks in smart grids. This method involves creating models of smart grids using both real-time streaming measurements and static data like system parameters and substation configurations. Based on the system model, estimation-based detection approaches have been utilized by the researchers.

In the static estimation method, each step of estimation is handled independently without any information being passed to the next step. The main approach utilized for identifying attacks is the Weighted Least Squares (WLS) estimation technique. The work in [34] employed WLS to detect failures that could lead to changes in network topology or measure incorrect voltages. Furthermore, the work in [35] examined the repercussions of FDIA and incorporated WLS into their detection methodology. To enhance the convergence speed, a recursive version of WLS was proposed in [36], which updates the state estimation using historical states. In [37], the WLS is applied to identify anomalies, in voltage controllers within the transmission system. The major drawback of using the WLS estimator approach is its dependence on the assumption that a smart grid operates in a steady-state with sufficient redundancy. However, in real-life scenarios, the smart grids encounter changes in demand and generation, making it challenging to maintain a stable condition [38].

In dynamic state estimation approaches, the methods such as Kalman filter (KF) are widely employed. The KF method entails forecasting the state based on the previous one and adjusting the prediction using measurements obtained at that moment. In [39], KF was utilized to identify FDIA in automatic generation control (AGC) systems emphasizing

the impacts of FDIA. Furthermore, in [40] an approach using KF for real time operations to estimate and detect FDIA was proposed. A significant limitation associated with Kalman filter is that it becomes more complex as the number of buses in a network grows because of the need to create Jacobian and error covariance matrices. Also, if there is nonlinearity in the network, the Kalman filter will struggle to accurately estimate and detect the cyberattacks [41].

2.2.2. Machine Learning Based Approach

The work in [42] investigated an Intelligent Remedial Action Scheme (IRAS) that aimed to distinguish between the cyberattacks and the physical disturbances in the smart grid. The approach utilized an anomaly detection technique based on decision trees with voltage and current phasors serving as the features. However, it is important to acknowledge the limitations of this method as the classification model could potentially be prone to overfitting, leading to false positives. Furthermore, relying on differential features of voltage and current phasors may not provide sufficient resilience against cyberattacks that can effectively hide their activities and evade detection.

In a study conducted in an IEC 61850 substation environment, a behavior-based intrusion technique proposed in [43] was implemented. The aim is to detect anomalies using dynamic features and acquire Generic Object Oriented Substation Event (GOOSE) and Manufacturing Message Specification (MMS) factors. For the experimental set-up, 261 normal traffic scenarios were randomly selected. The Packet Capture (PCAP) files containing the most likely IEC 61850-related vulnerabilities based on 27 attack scenarios were inserted randomly into the dump file for analysis. However, the study only focused

on detecting a GOOSE spoofing attack using dynamic features and disregarded the other types of attacks.

The work in [44] introduced a rule-based Network Intrusion Detection System (NIDS) for digital substations. The Rule based methods typically involve analyzing data to identify patterns and establish classification rules. However, this can be quite challenging when working with large datasets that have numerous features leading to complex models and potential problems with overfitting. Additionally, the NIDS discussed in the study did not take into account the disturbance scenarios, which were consequently omitted from the classification analysis.

Hyunguk et al. [45] proposed an anomaly detection model through normal behavior profiling of the MMS and GOOSE packets in order to identify abnormal events in the network. According to Figure 2.1 during the processing stage, MMS and GOOSE packets are extracted from the data collected in the substation network using packet filtering. These packets are then grouped into datasets using a 3-phase preprocessing technique. To ensure accuracy either EM (Expectation Maximization) or LOF (Local Outlier Factor) was applied to identify and remove any outliers from these datasets. Once the outliers were removed, the normal-behavior models for each of the three datasets using a one class SVM algorithm was created. In real time, the anomaly detection engine receives packets for preprocessing and compares them against the established normal behavior models to determine whether they are within expected parameters or exhibit abnormal behavior. Based on this comparison an alarm and log are updated accordingly.

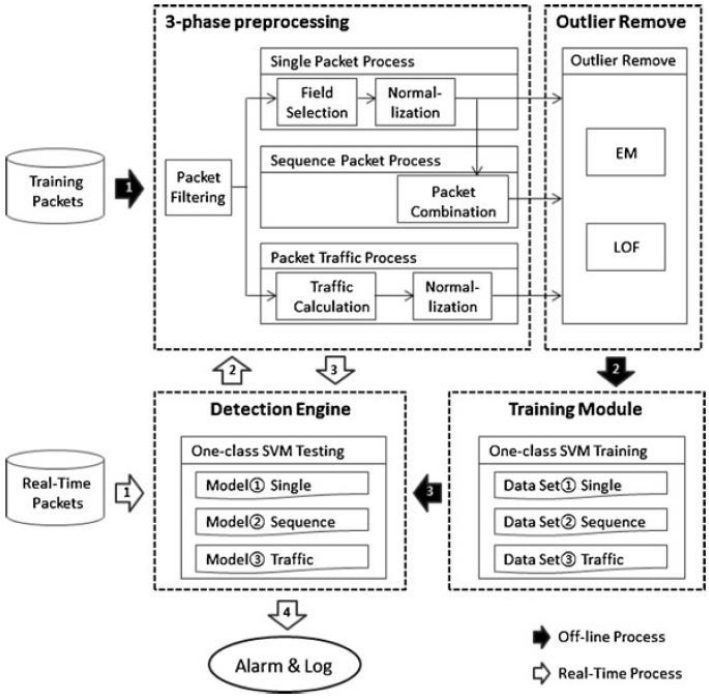


Figure 2.1: Architecture of the Anomaly Detection System through Normal Behavior Profiling [45]

In [46], a collaborative intrusion detection system (CIDS) that can be executed within several IEDs is presented. The algorithm-based intrusion system is used to detect intrusions in both GOOSE and Sample Value (SV) messages. A significant drawback in the implementation of this approach is that such IDS will require a large amount of communication between devices to facilitate effective detection and mitigation of cyber-attacks. This high level of communication can result in substantial network congestion when more than two CIDS IEDs are attacked simultaneously, which may, in turn, compromise the system's overall performance and reliability.

In [47], a hybrid intrusion detection system (HIDS) was proposed. The purpose of this system was to learn and analyze the behavior of power systems during situations such,

as disturbances, normal control operations and cyber attacks. To implement and generate data for the power system scenarios, a hardware in the loop testbed was utilized. This testbed included a real-time digital time simulator (RTDS) that simulated transmission lines, breakers, generators and load. The hybrid IDS employed a common paths mining approach, which proved to be effective in classifying 90.4% of the tested scenario instances. However, it is important to note that this approach relies on detecting correlations or similarities in the patterns of system activity that may not indicate malicious intent. Consequently, there is a possibility that legitimate user actions might be flagged as threats by the system leading to an increase in false positives.

The work in [48] introduced a concept referred to as True Data Integrity. It presents an approach using an Agent Based Model to measure the vulnerability of data to attacks. The True Data Integrity-Agent Based Model (TDI-ABM) focuses on analyzing time series values and comparing them against malicious values at a specific moment. By evaluating the values, the model predicts the subsequent values and calculate any error at the end of the timeframe to determine if an attack has taken place. Through experimentation with replay attacks using the Artificial Feed forward Network (AFN) the model achieved a 98.19% accuracy in detecting false data. However, it is worth noting that this study did not investigate scenarios involving disturbance events. Additionally, training this model with AFN can be computationally expensive and time consuming, particularly when dealing with large datasets. These factors may limit its scalability and efficiency in real world applications where fast and precise predictions are often necessary.

Zia et al. in [49] proposed a method for detecting FDIA and locating compromised meters using machine learning techniques such as binary relevance (BR) and classifier

chain (CC). The BR method involves training one binary classifier for each label using a large set of synthesized measurements. These classifiers are then used for testing purposes. The CC method trains multiple binary classifiers that are interconnected in a chain based on the feature space. The findings of this research demonstrate that the BR method achieves an accuracy rate of 95.1% in detecting and locating FDIAs surpassing algorithms, like CC, SVM and light gradient-boosting machine (LGBM). Nevertheless, this study did not explore the adaptability of these methods to other cyberattacks or systems.

2.3. Previous Work on Detection and Classification of Cyberattacks

This section provides an overview of the existing methodologies that have been introduced in the literature for the detection and classification of cyberattacks. The previous works are classified into two different types: non-machine learning based approach and machine learning based approach as discussed below.

2.3.1. Non-Machine Learning Based Approach

Several studies have utilized non-machine learning approaches such signature-based techniques to detect and classify cyberattack threats and anomalies. Signature-based approaches rely on pre-existing databases and fixed signatures to detect and classify attacks that are known [50].

A method for detecting and classifying cyber attacks was proposed in [51], which leverages Gaussian processes to identify anomalies, within different attack types. Furthermore, the work in [52] presented a cyberattack detection and classification technique that relies on the Pearson correlation coefficient to measure the relationship between parameters of Phasor Measurement Units (PMUs).

The work in [53] presented a set of fifty signature rules for Modbus protocols used in serial communication interfaces. The study employed the Snort IDS to verify these rules. Nonetheless the study provides instructions, on how other IDS systems can adopt these rules. Each rule encompasses a text field, which incorporates protocol specific details. However, it is important to mention that the study did not offer any numeric results regarding the efficiency of these rules.

The study in [54] focused on the DNP3 protocol. The work also utilized the Snort IDS to provide signature rules. A template was established for intrusion detection, which was then utilized to create signature rules for the DNP3 protocol. These generated signature rules are capable of detecting and classifying anomalies in the protocol including reconnaissance attacks, DoS attacks and hybrid attacks. However, the study did not provide any evaluation procedure in their work.

The study in [55] introduced an IDS framework for substations, which employs signatures and focuses on the active power limitation attacks. They developed a stateful analysis plugin, which can be incorporated into an IDS. The plugin has three functions: it decodes the application layer packets, it applies rules for detecting attack patterns and differentiates between content conditions and state tags, and it updates the protected devices states. The study tested this plugin using the Manufacturing Message Specification (MMS) protocol per IEC 61850 standards, detected and classified two attacks but did not provide numerical results.

The signature-based technique is reliable and has a low rate of false positives. However, it cannot detect unknown attacks that are not specified by any signature. This limitation results in various intrusion detection system (IDS) topologies [50].

2.3.2. Machine Learning Based Approach

In the study presented in [56], various machine learning models were assessed in a system that compared SCADA and IEDs. The findings revealed that the JRipper + Adaboost algorithm demonstrated a low false positive rate when employing a three-class classification system (normal, disturbance and attack categories). However, the algorithm encountered challenges in differentiating between specific types of attacks, such as remote tripping attack, relay setting change attack and FDIA. As a result, it exhibited a high rate of false positives.

M. Keshk et al. in [57] introduced a method known as privacy preservation intrusion detection (PPID) to identify intrusion events in SCADA systems. However, the findings of the study indicate that prioritizing privacy preservation may result in information unavailability for intrusion detection, which potentially impacts the accuracy of detecting the different types of attacks.

In [58], a two-layer machine learning model that relied on a Random Forest Classifier (RFC) was proposed. The main aim of the first layer was to distinguish between normal operation and cyberattacks. Subsequently the second layer categorized the identified state into types of cyberattacks. However, the RFC approach used in their study was simplistic as it did not involve tree pruning or any stopping criteria, which made the classification model susceptible to overfitting. Consequently, the approach have high misclassification errors when used on the foreign test datasets.

In [59], a sequential classification machine learning model known as bidirectional long short-term memory (BiLSTM) was utilized. The BiLSTM model demonstrated effectiveness in identifying FDIA and replay attacks with a low rate of false negatives at

0.372%. However, this approach is time and resource consuming when applied to large datasets due to its complexity. Furthermore, the study did not thoroughly explore how the selection of datasets, for training and testing impacts the accuracy of classification.

In [60], a classification model which is made up of enhanced Extra Tree (ET) that utilizes the Synthetic Minority Oversampling Technique (SMOTE) was developed. The aim of this model is not to detect attacks but also to identify the specific types of attacks. By using SMOTE, the study addressed the challenge of imbalanced data by oversampling the minority class. Additionally, the study employed the ET classifier, which is a tree-based ensemble method specifically designed for dealing with unbalanced classification problems. The experimental findings indicate that their proposed ET-SMOTE algorithm surpasses existing benchmark models, in terms of accuracy achieving an accuracy rate of 99.79%. However, it is worth noting that this study solely relied on a single dataset, which may introduce some vulnerability to overfitting issues.

2.4. Research Gaps

The previous work that has been published in the literature is summarized in Table 2.1. The following is a summary of the identified research gaps from the previous work.

- The literature review revealed that there is a lack of research focusing on identifying and classifying the types of attacks and disturbance scenarios. Moreover, the cyberattack datasets tend to be complex and imbalanced due to the rarity of attack scenarios. This underscores the importance of developing a detection and classification model.

- The previous work published in the literature has indicated the necessity for a technique that can distinguish between different types of attacks and highlight the relevant features associated with each type.
- The literature has also revealed the need for a technique that is not susceptible to overfitting and is not sensitive to the selection of training and testing datasets.

Table 2.1: Literature Review Table of Detection and Classification of Cyberattacks in Smart Grids

Ref #	Detection Technique		Classification	Overfitting	Examining various datasets	Data set Interchange
	Machine Learning	Non-Machine Learning				
[34]-[40]	×	√	×	×	×	×
[42]	√	×	×	√	×	×
[43]	√	×	×	**NA**	**NA**	×
[44]	√	×	×	×	×	×
[45]	√	×	×	√	√	×
[46]	√	×	×	**NA**	**NA**	√
[47]	√	×	×	√	×	×
[48]	√	×	×	×	×	×
[49]	√	×	×	**NA**	×	×
[51]-[55]	×	√	√	×	×	×
[56]	√	×	√	×	×	×
[57]	√	×	√	×	×	×
[58]	√	×	√	×	√	×
[59]	√	×	√	×	×	×
[60]	√	×	√	×	×	×

NA – Not Available, × - Not Performed, √ - Performed

2.5. Summary

This chapter summarizes the previously published work in the literature related to the detection and classification of cyberattacks in smart grid. Initially, the chapter highlighted the research in the area of cyberattack detection, then it discussed the approaches that have been employed in identifying anomalies in the smart grid. Secondly, the chapter presented a summary of the approaches that have been utilized to detect and classify different cyberattack types. The studies that presented several approaches for the detection and classification of cyberattacks have been reviewed and the main outcomes of the studies were highlighted. In addition, the limitations of these studies were presented and discussed. The identification of research gaps and the subsequent discussion of the recommended approach to address these gaps will be the focus of this thesis. The next chapter explains the common machine learning approaches in smart grid used to compare with the proposed approach.

3. Machine Learning Methodologies used in Smart Grid

3.1 Introduction

The field of machine learning is a significant aspect of artificial intelligence. It has proven to be valuable when it comes to managing the amount of data generated by smart grids. Machine learning techniques have become tools, for analyzing data and making decisions that ensure the smooth operation of the grid. Through machine learning, the information is gained from raw data and predictions are made based on that information. This involves utilizing algorithms that carefully examine data using a set of instructions to generate predictions and make informed decisions. In the context of smart grids, machine learning functionalities encompass tasks, like power generation management, optimizing schedules, determining prices, detecting faults or malfunctions, predicting consumption patterns, implementing adaptive control measures as well as identifying and classifying cyberattacks. Integrating machine learning into the smart grid is essential because of the incorporation of new technologies into the grid. This will play a role in the much-needed task of safeguarding the grid against the rising number of cyberattacks. In this section the background of the common machine learning techniques used in this thesis is provided.

3.2. Machine Learning-Based Algorithms

One effective approach, for identifying and categorizing cyberattacks in grids involves the use of machine learning techniques. Unlike other algorithms, machine learning relies on data from the system being analyzed. There are two types of machine learning algorithms: supervised and unsupervised. Supervised learning utilizes labelled datasets to classify data or make predictions about outcomes. On the other hand, the unsupervised learning uses unlabelled data to uncover patterns for clustering or association purposes [61]. This thesis

specifically focuses on the implementation of machine learning methodologies as the dataset used is labelled to enable the identification and classification of cyberattacks in the substation automation systems.

3.3. Supervised learning Methods

In supervised learning, the model requires labeled data to learn the patterns. Each input is linked to a specific output as (s_i, y_i), where s_i represents the i^{th} input sample and y_i is the label that falls under normal, disturbance, or attack type [62]. In this thesis, the supervised learning methodologies compared with the proposed approach are Decision tree, Support Vector Machine and K-nearest neighbor.

3.4. Decision Tree Algorithm

A decision tree is a machine learning method that has a structure resembling an inverted tree or pyramid. It is employed to address issues related to event classification in smart grids. In this technique each internal node of the tree represents an input feature and the branches connecting these nodes are determined by input characteristics. The resulting values of the output feature are assigned along these branches [63]. Decision trees visually depict the decisions that need to be made, potential outcomes and various combinations of decisions and events as shown in Figure 3.1 [64].

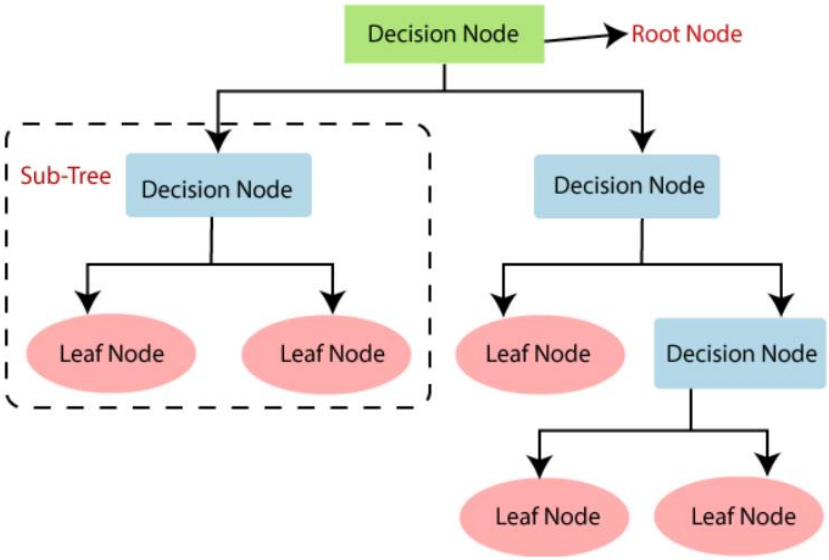


Figure 3.1: Decision Tree[61]

There are algorithms that can automatically generate a decision tree from a dataset. These algorithms include Iterative Dichotomiser 3 (ID3), Classification and Regression Trees (CART), J48, C4.5, C5.0 Chi square Automatic Interaction Detector (CHAID) and Quick, Unbiased, Efficient Statistical Tree (QUEST). In this research, the CART algorithm is used to construct the decision tree because it can handle both classification and regression tasks. To predict the class of the dataset used in this research, the algorithm begins at the root node of the tree. It compares the values of the root attribute with those of the test system dataset attribute. It then proceeds along the branches based on this comparison and moves to the next node. This process is repeated for each node by comparing attribute values until the leaf node of the tree is reached.

3.4.1. Decision Tree Attribute Selection Measure

The decision tree is grown by selecting the optimal split among the attributes of the datasets, based on the Gini index that measures the impurity of the tree nodes [65]. The

attribute that is selected for splitting is determined by measuring the node impurity and selecting the attribute with the lowest weighted Gini Index. The pure node has a characteristic of all the observation being from the same class [66].

$$Gini(t) = 1 - \sum_{i=0}^{c-1} [p(i|t)]^2 \tag{3.1}$$

Where $p(i|t)$ is the portion of observation that belongs to class i at a given node *t* and c, the number of class labels. The Weighted Gini index $\left(Gini_{weight}\right)$ is defined as:

$$Gini_{weight} = \sum_{t=0}^{n} \frac{X_t}{T} \times Gini(t) \tag{3.2}$$

Where, X_t is the number of scenarios in node *t*, T is total number of scenarios, $Gini(t)$ is the Gini index value at a given node *t*, and *n* is the number of nodes.

3.4.2. Strengths and Limitations of Decision Trees

The decision tree methodology is an effective technique for the classification of events in several applications. Below are the advantages and disadvantages of this methodology.

Advantages of the Decision Tree Algorithm

- The process of using this method is similar to how humans make decisions in real-life.
- This approach proves to be helpful in solving decision-related problems, while considering all possible outcomes.
- It requires less data cleaning in comparison to other algorithms.

Disadvantages of the Decision Tree Algorithm

- The decision tree can be complex when it contains numerous layers.
- It can be susceptible to overfitting [67].
- If there are more class labels, the computational complexity of the decision tree may also increase.

3.5. Support Vector Machine Algorithm

In the field of machine learning, the Support Vector Machine (SVM) is also a popular supervised learning method. It helps to classify or regress data sets that can be either discrete or continuous depending on the type of data [68]. The SVM algorithm creates a linear classifier by assigning training instances to predefined categories. Its primary objective is to find the line or decision boundary that separates classes in an *n*-dimensional space so that new data points can be accurately assigned to their appropriate category. This optimal decision boundary is known as the hyperplane of the SVM.

3.5.1. Hyperplane and Support Vectors in the SVM algorithm

In SVM, there can be one or multiple hyperplanes that separate classes in an *n*-dimensional space. The number of dimensions in the hyperplane depends on the features in the dataset. For instance, if there are three features, the hyperplane will be a 2-dimensional plane as shown in Figure 3.2. Three lines divide the three classes into their groups. The hyperplane is created with the maximum margin, which represents the distance between data points. The data points or vectors that are the closest to the hyperplane and have an influence on its position are called Support Vectors. They are named so because they support the hyperplane.

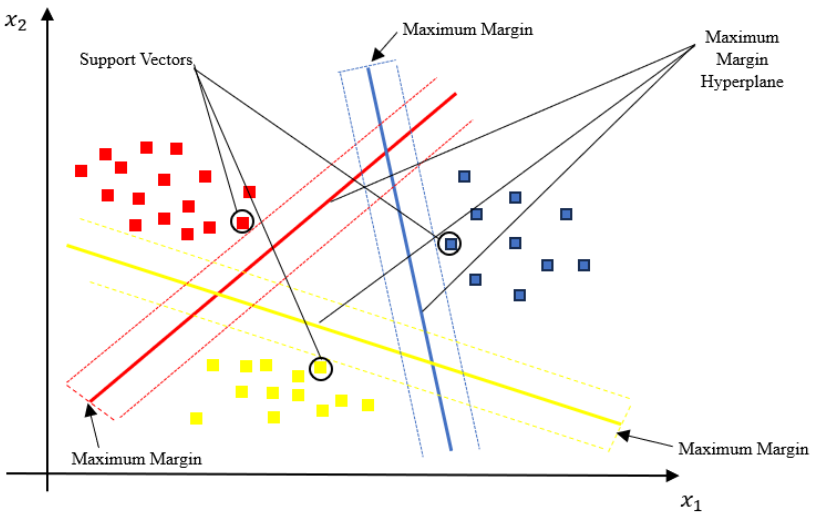


Figure 3.2: Representation of SVM in a 2-dimensional space [68]

To reduce the computational complexity, kernel functions are utilized to represent the data mapping. In this research, a Gaussian kernel is employed for SVM due to its ability to classify data based on statistical variances with high computational efficiency, owing to its nonlinear properties. The Gaussian kernel is mathematically defined as follows:

$$K(x_i, x_{i'}) = exp\left\{-\gamma \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2\right\} \tag{3.3}$$

Where, γ is the kernel coefficient. The SVM algorithm will undergo accuracy testing through cross-validation, with varying penalty parameters denoted by C, and kernel coefficients γ.

3.5.2. Strengths and Limitations of Support Vector Machine Algorithms

Using SVM for identifying events in substation automation systems can also be considered as it offers significant advantages. Below are the benefits and drawbacks of this methodology.

Advantages of SVM Algorithm

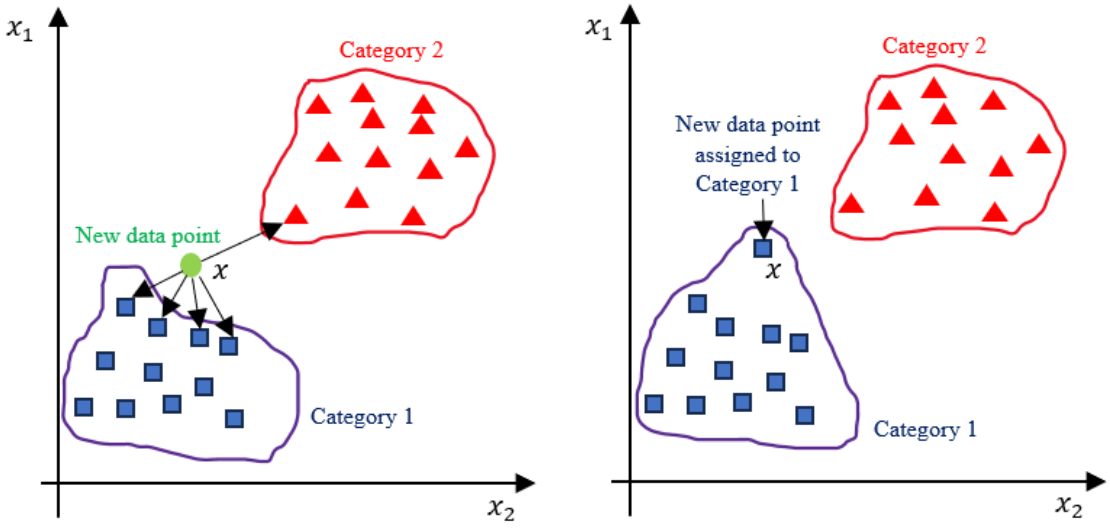
- It excels in handling high dimension data.
- SVM is particularly useful for small datasets as it exhibits good generalization abilities.
- With the use of kernel functions, it can effectively classify data.

Disadvantages of SVM algorithm

- One of the drawbacks of the SVM algorithm is that it struggles to handle large datasets efficiently.
- The methodology is computational expensive and takes large training time.

3.6. K-Nearest Neighbor Algorithm

The *K* Nearest Neighbor (KNN) algorithm is an effective technique in machine learning. It has shown its usefulness in tasks, including fault detection, localization and classification [69]. It works by looking at the class that is commonly chosen by the neighbors of an object. It is referred to as a lazy learner algorithm because it stores all training samples and only builds a classifier when a new, unlabeled sample needs to be classified [70]. Figure 3.3 provides an illustration of how KNN works. It relies on learning through resemblance by comparing test samples with training samples that are similar to them. To classify a data point using KNN, the algorithm searches for its K-nearest neighbors and measures the distance to each neighbor. It then counts how many data points belong to each category among these neighbors. The assigned class label is determined by which category has the majority of the neighbors, which in this instance is category 1.



(a) (b)

Figure 3.3: Implementation of the Algorithm for K=5 (a) Before KNN and (b) After KNN [71]

3.6.1. Selecting K in the KNN Algorithm

The effectiveness of the K-nearest neighbor algorithm is influenced by the choice of "K" [72]. However, determining the value for "K" when implementing the KNN algorithm on a dataset is not straightforward. To find the "K" value multiple values are tested to identify the optimal value. If "K" is too small there is a risk of overfitting due to noise in the training dataset. On the other hand, if "K" is too large, misclassification may occur as distant data points could be included in its neighbors list.

In this research, the K-nearest neighbor (KNN) algorithm is utilized for data classification by determining its closest k neighbors. The proximity between the data points is measured using the Euclidean distance equation.

$$d_{ij} = \|s_i - s_j\|, s_j \in S \tag{3.4}$$

In the context of this study, the symbols *S* and *s* represent the labeled and unlabeled data, respectively. When the value of *k* exceeds 1, the classification of data is determined by the majority of its neighboring data points. To determine the optimal *k* value, several different *k* values will be evaluated through a process of cross-validation, with the goal of maximizing the accuracy of the classification.

3.6.2. Strength and Limitation of KNN Algorithms

Advantages of the KNN Algorithm

- This method is easy to comprehend and implement.
- It has the ability to handle noisy training data.
- Performs well in scenarios where a single sample may have multiple class labels [72].

Disadvantages of KNN Algorithm

- When dealing with a large number of potential neighbors to compare with an unlabeled sample, the computational costs can be quite high [72].
- It is sensitive to the local structure of the data and has memory limitations [73].
- Due to being a supervised lazy learner, it may also run at a slower pace.

3.7. Summary

The classification process can be divided into two phases. The first is the training phase, where the classification model is built. Then there is the classification phase, where the trained model is used to assign an unknown data object to one of the predefined class labels. In this section, different commonly used classification techniques in data mining were explored. A study of algorithms such as Decision Trees, Support Vector Machines (SVM) and K-Nearest Neighbor (KNN) were discussed. The strengths and weaknesses of each algorithm were also highlighted. By examining the pros and cons of each method, this thesis offers a framework, for comparing the proposed methodology with these classifiers in detecting and categorizing cyberattacks.

4. Analysis of the IEC 61850 Substation Communication Standard

4.1.Introduction to IEC 61850 Standard

In the past, substation automation systems utilized master/slave architectures, which relied on communication protocols such, as Modbus and Distribution Network Protocol (DNP) to transmit substation data to a remote location. These protocols operated based on tags requiring users to access data by providing a tag or an index number. While this approach ensured a dependable communication network, the engineering process required to implement these protocols complicated the entire system.

The advancement of microprocessor technology and data networking has led to the adoption of Ethernet-based systems as the preferred method of communication in IEDs, surpassing serial communication. This preference for Ethernet offers benefits such as reduced wiring time, lower cabling costs and improved network addressability. However, the main drawback is the utilization of data protocols in non-standardized systems, which hinders interoperability between IEDs, from different vendors [74]. As a result, such substations require the use of complex protocol converters.

The development of the IEC 61850 standard was driven by the necessity to address issues related to interoperability and interchangeability [75]. Introduced in 2004, the IEC 61850 is a standard that integrates various practices for substation automation [76]. It incorporates the utilization of logical nodes and offers well-defined procedures for designing, modelling, representing data and configuring IEDs within a substation automation system. This has resulted in enhanced interoperability among IEC 61850 complaint IEDs compared to those that do not comply [16]. Notably, the IEC 61850 standard has gained adoption across electrical substations worldwide [77].

Table 4.1: Scope and Outline of the IEC 61850 standard [78, 79]

Part Number	Title
IEC 61850-1	Introduction and overview
IEC 61850-2	Glossary
IEC 61850-3	General Requirements
IEC 61850-4	System and Project Management
IEC 61850-5	Communication Requirements for Function and Device Models
IEC 61850-6	Substation Automation System Configuration Language
IEC 61850-7	Basic Communication Structure for Substation and Feeder Equipment
IEC 61850-7-1	Principles and Models
IEC 61850-7-2	Abstract Communication Service Interface
IEC 61850-7-3	Common Data Classes
IEC 61850-7-4	Compatible Logical Mode Classes and Data Classes
IEC 61850-7-5	Technical Report
IEC 61850-8	Specific Communication Service Mapping (SCSM):
IEC 61850-8-1	Guideline For Mapping from IEC 61850 To IEC 60870 5-101/-104 (Technical Specification)
IEC 61850-9	Process Bus Mapping
IEC 61850-9-1	Sample Values (SV) Over Serial Uni-directional Multi-Drop Point-To-Point Links
IEC 61850-9-2	Sampled Values Over ISO/IEC 8002-3
IEC 61850-10	Conformance Testing

The IEC 61850 standard consists of ten parts and multiple subsections that encompass aspects concerning data modeling and communication framework. Table 4.1 highlights each parts scope and relevance, to substation automation systems.

4.2. Communication Architecture of IEC 61850

According to the IEC 61850 standard an Ethernet-based substation automation structure follows an approach consisting of three levels: station level, bay level, and process level. Furthermore, the architecture incorporates two types of buses: process bus and station bus. Figure 4.1 illustrates the IEC 61850 substation architecture.

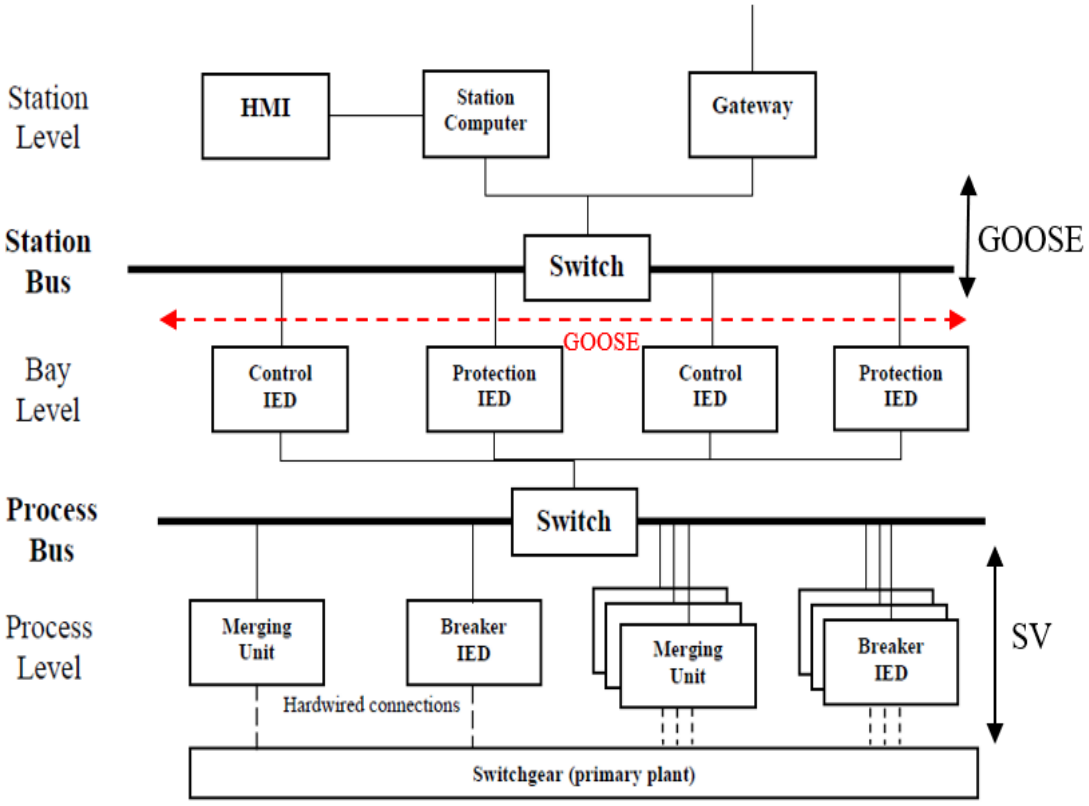


Figure 4.1: Architecture of an IEC 61850 based substation

The Process level consists of merging units (MU), sensors, Resistance Thermal Detectors (RTDs) and breaker IEDs to link switchgear equipment together with the substation automation systems located in the Bay level kiosks [75]. To facilitate communication and data transfer between the process level and bay level, the process bus acts as a conduit. It enables the transmission of raw data including measurements from current and voltage transformers as well, as control information [75] .

The Bay level serves as a link facilitating the connection of different control and protection IEDs using station level ethernet switches. To ensure the separation of substation components like lines and transformers from the remainder of the substation, serial connections are employed [78]. Additionally, all automation systems located at the Bay level are housed in separate kiosks ensuring distance from the switchgear equipment [80]. Communication between the Station level and Bay level and even inter IED communication within both levels is made possible through the station bus.

At the station level, there is Human Machine Interface (HMI), station computers, a database, and remote communication interfaces. These tools are used to archive, automate, store data, and manage multiple Bay level devices with the help of specialized software.

4.3. IEC 61850 Information Model

This section describes the storage of data and metadata in an IED and their representation in the IEC 61850 standard. The model consists of elements that describes the information model such as setpoints, measured values, and sequence of events. Additionally, the components related to the communication configuration, which is referred to as the information exchange model in IEC 61850-7-1 subsection is described.

4.3.1. Logical device

The data source in IEC 61850 begins with a server that stores files and is connected to a physical device where a logical device operates. Within this logical device, there are various logical nodes [79].

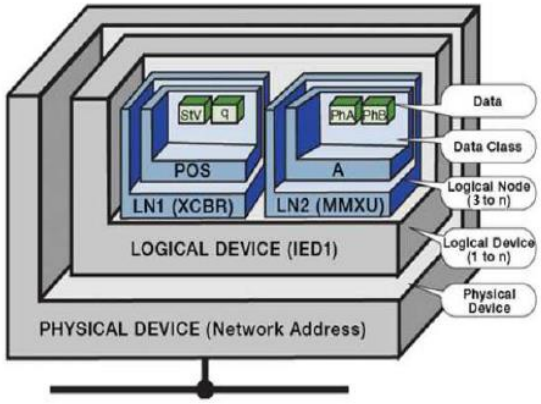


Figure 4.2: IEC 61850 device representation [81]

A physical device acts as a means of communication, between the logical devices such, as Ethernet or other networks. In the logical device (LD) model there are nodes that provide the required information for a device. The LD outlines the functions that need to be carried out by a device as shown in Figure 4.2. A logical device comprises of logical nodes and each device must have at least three logical nodes [82].

4.3.2. Logical Nodes

In the IEC 61850 Standard, the logical node (LN) holds significant importance. LNs act as virtual representations of the fundamental functions within a SAS and serve specific functions through predefined groupings of data objects[83]. Table 4.2 presents the complete list of logical node groups and their respective labels.

Table 4.2: Categorisation of Logical Nodes in IEC 61850

Logical Groups	Group Labels	Logical Groups	Group Labels
Automatic Control	A	Protection Related	R
Supervisory Control	C	Sensor and Monitoring	S
Generic	G	Instrument Transformers	T
System logical nodes	L	Switchgear	X
Metering	M	Power Transformers	Y
Protection	P	Further Power System Equipment	Z
Interfacing and archiving	I		

To simulate a complete device, LNs can function as building blocks. From Figure 4.2, examples of LNs include the XCBR for LN1, which portrays circuit breaker capabilities of a switch, and the MMXU for LN2, which provides all electrical metering measurements in 3-phase systems such as voltage, current, watts, vars, power factor, etc. [78]. Table 4.3 shows all the logical node classes in an IEC 61850 based substation and their description [84].

Table 4.3: Logical Node Classes in IEC 61850 substation

LN Classes	Description	LN Classes	Description
GGIO	Generic logical node	PTOC	Time overcurrent protection
MMTR	Metering	RBRF	Breaker failure
MMXU	Measurement unit	XCBR	Circuit breaker
PDIF	Differential protection	XSWI	Circuit switch
PDIS	Distance protection	YPTR	Power transformer

4.3.3. Data Class

Data Class or Data Objects (DOs) refer to preassigned names given to objects that are associated with one or more nodes. Each logical node is linked to one data object. Common data classes serve purposes, such, as indicating integer status, measured values or defining analog settings [85]. For instance, the OpCnt data object of the logical node class XCBR (Circuit Breaker) denotes the operation count of the circuit breaker. The OpCnt utilizes the integer status data object class. Table 4.4 showcases the data objects and their respective functions in substations based on IEC61850 [80].

Table 4.4: Data Objects in IEC 61850[86]

Data Class Name	Description
A	Phase to ground amps
Ang	Angle between phase current and voltage
BlkCls	Status Information
Loc	Local operation
Operation of a logical node	Op
Pos	Switch position
Str	Starting of a logical node
Tr	Trip activation

4.3.4. Data Attribute

Lastly, the data objects, which holds data that has certain features referred to as Data attributes. Data attributes are predetermined attributes that can be used by numerous objects, like value, quality, timestamp, and description. These shared attributes are outlined in Clause 6 of IEC 61850-7-3 [87]. Common Data Classes (CDC) are standard groups of data attributes, as defined by IEC 61850. Every data object within a logical node is part of

a CDC. The data attributes consists of parameters such as Boolean, Coded Enum, integer, Bit String, and floating point, that make up the data types [80].

4.3.5. Naming Convention in IEC 61850

The IEC 61850 standard uses a hierarchical naming convention for devices, logical nodes, data objects, and data attributes. This naming convention is crucial because it eliminates ambiguity. As shown in Figure 4.3, the first part of the naming convention is the logical device name chosen by the utility. The second part refers to the logical node. As explained in Section 4.3.2, the first letter of the logical node represents the logical group to which it belongs. In the figure, the logical node begins with "M", which stands for metering. The third part indicates the instance number of the logical node, which in this case is "Feeder number 3". The fourth part refers to the "Data Object Name", which is defined as Phase-to-Ground amps. Finally, the fifth part is the Data Attribute of the logical node defined in a CDC. The *PhsA* represents Phase A, *cVal* represents the complex value, *mag* is the magnitude of the complex value, and *f* is the floating-point value.

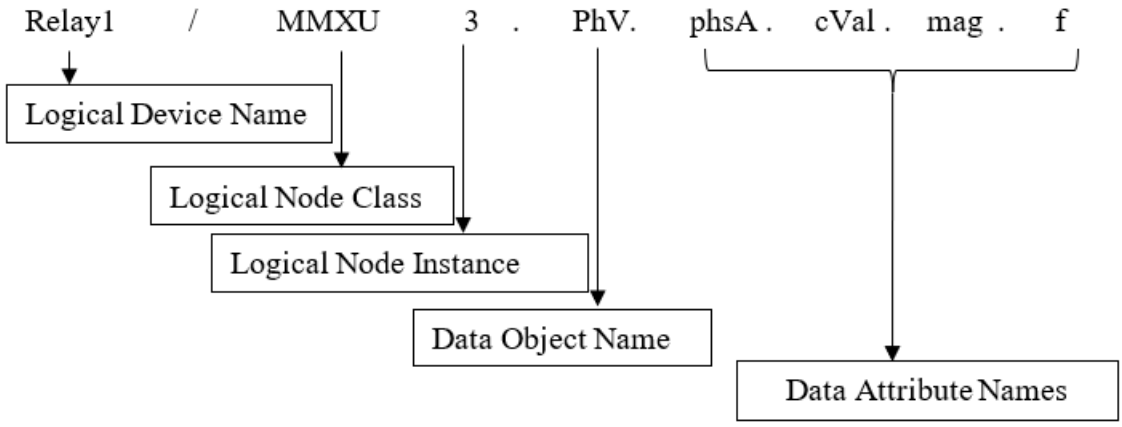


Figure 4.3: Default Naming scheme of IEC 61850

4.4. Substation Configuration Language

Before the implementation of the IEC61850 protocol, communication between vendor IEDs was limited [85]. This was due to manufacturers designing their products using proprietary tools, making it impossible to achieve interoperability. To resolve this issue, the Substation Configuration Language (SCL) was introduced in IEC 61850-6. The SCL is an Extensible Markup Language (XML) based language that is used to describe how different vendor IEDs connect and interact with each other. The SCL enables the exchange of relevant information about both the entire system and individual components [88]. The SCL allows vendors to define the functionalities of IEDs allowing users to conveniently set up IEC 61850 clients without the need for a list of data points. Also, it facilitates the export and import of IED configurations to applications and tools.

The IEC 61850-6 introduces two tools for substation automation system configuration: the IED Configurator and the System Configurator. The IED Configurator, often vendor-specific, creates and loads IED configuration files. Conversely, the System Configurator, vendor-independent, merges various IED configuration files into one substation-wide file, which then guides specific IED configurations. The different types of SCL files illustrated in the configuration information flow process in Figure 4.4 are described below [88]:

- System Specification Description (SSD): Describes the entire single-line diagram system and the individual device functions.
- Substation Configuration Description (SCD): Describes a single substation automation system's communication and function configuration.
- IED Capability Description (ICD): Describes the complete communication functions and data model capabilities supported by an IED.

- Configured IED Description (CID): Describes all the data required from the system to configure a specific IED.
- System Exchange Description (SED): Describes the Information required for data exchange between substations.
- Instantiated IED Description (IID): Describes the configuration of an IED for a specific project.

In this research, the IID file was accompanied with the data set files used to evaluate the proposed approach.

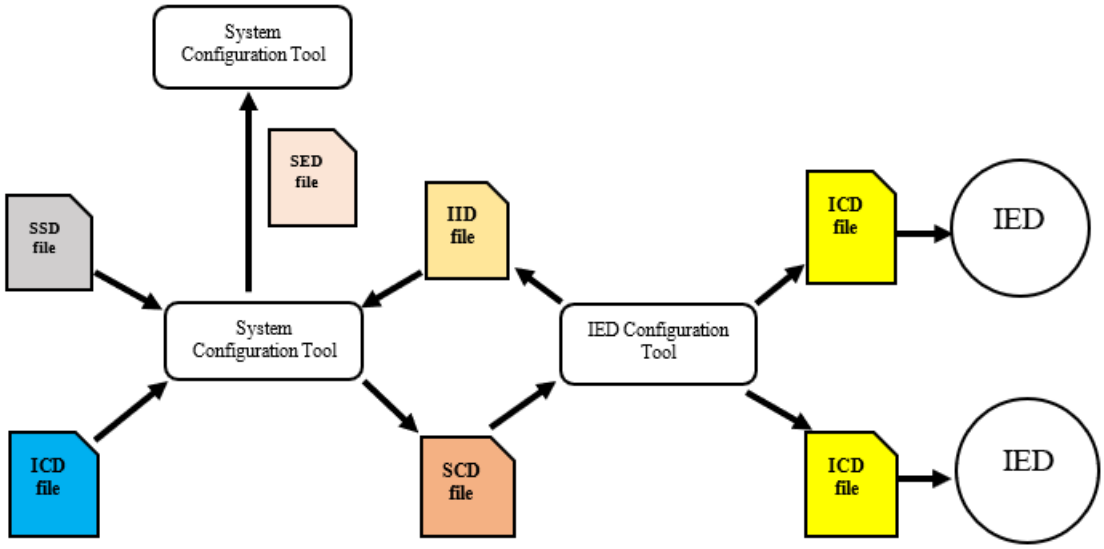


Figure 4.4: Information flow of the configuration process

4.5. Overview and Configuration of GOOSE and Sampled Values in IEC 61850

Based on the guidelines provided by IEC 61850, smart grid application messages can be divided into two categories: Subscriber/Publisher and Client/Server. The Subscriber/Publisher messages serve time-critical purposes like sending control commands such as tripping, blocking or indicating state changes as well as performing metering and

protection functions. On the other hand, Client/Server messages are typically used for voltage control, condition monitoring and data recording in case of failures. In this thesis the focus is on the publisher/subscriber message category, which includes GOOSE and Sample values. These specific message types were utilized in this research.

4.5.1. IEC 61850 Publisher-Subscriber Architecture

In this system architecture, there are two kinds of devices that interact with each other; the publisher and the subscriber. One device function as a GOOSE publisher while another device acts as a GOOSE subscriber. The publisher device broadcasts messages to all devices on the network and only the subscriber device captures the message to access the data. To ensure that devices receive GOOSE messages within 3 milliseconds after an event in the substation, the publisher device increases the rate at which the messages are sent through a retransmission mechanism. Afterward the device maintains a pace of message sending allowing the subscriber device to identify any communication failures [89]. Figure 4.5 provides an illustration of how the publisher-subscriber communication model operates.

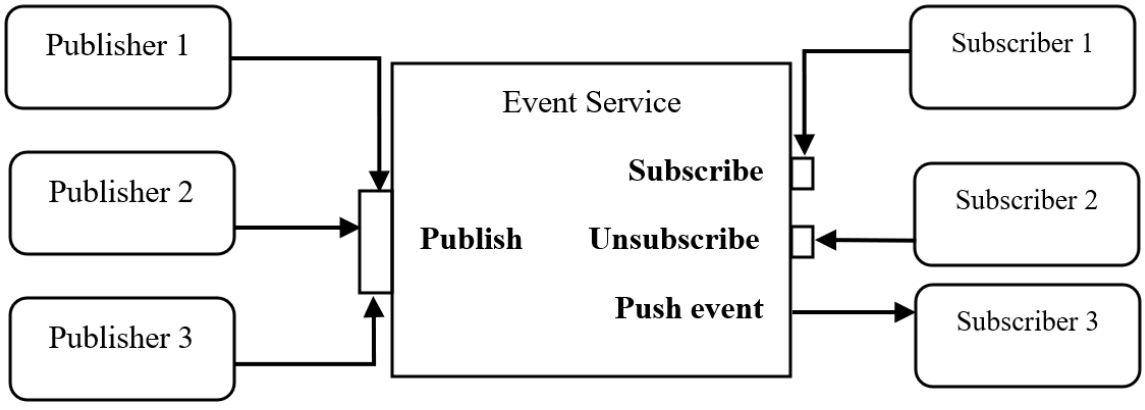


Figure 4.5: Publisher and subscriber communication [90].

4.5.2. Retransmission Mechanism

The retransmission mechanism process used by the publisher device is illustrated in Figure 4.6. In the two-state process, the first state is characterized by inactivity while the second state involves the occurrence of an event. During the first state, the retransmission mechanism is implemented to ensure that the same set of data is transmitted at intervals of Tmax. Although there may be instances where some retransmissions are lost or susceptible to errors during transmission but in the end the subscriber will eventually receive the correct information. In the second state when there are changes in the data, new information is retransmitted shortly after a brief interval of Tmin (where Tmin is less than Tmax). This sequence is repeated a few times, with each subsequent retransmission interval increasing until it matches the length of Tmax at which point the system reverts back to the first state. The retransmissions that take place during this state are referred to as "fast retransmissions" and the manufacturer has flexibility in determining the duration of these intervals [91].

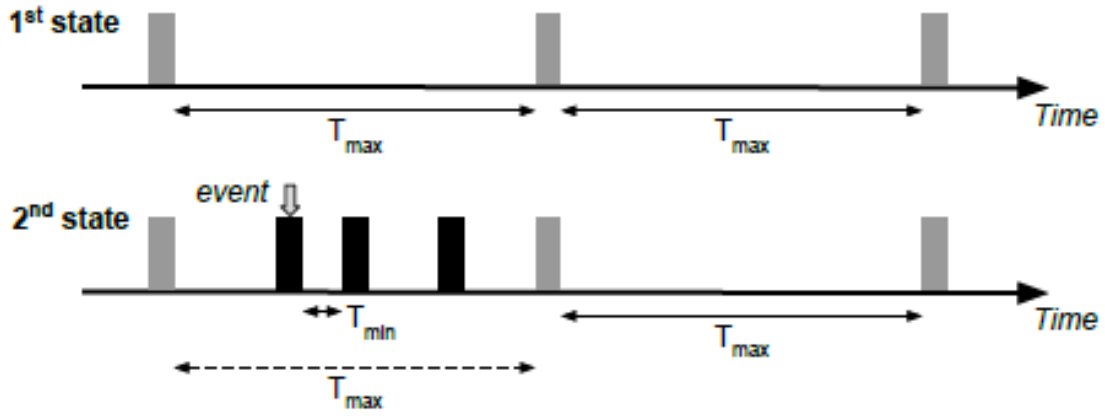


Figure 4.6: Retransmission mechanism in publisher-subscriber architecture [91]

4.5.3. Generic Object-Oriented Substation Events (GOOSE)

The GOOSE messages refer to a type of peer-to-peer communication that utilizes services to send the same event message to multiple IED devices in a substation. The purpose of GOOSE messages can vary depending on their application types. These types are categorized based on their time requirements for transmission as defined by IEC 61850 [92]. The GOOSE protocol's primary objective is to enable rapid and accurate data transmission between two or more IEDs. It achieves this by using Ethernets message design, which allows for sending of content without establishing a connection resulting in fast data transfer. However, one potential drawback of this approach is that GOOSE may be less reliable since it doesn't provide confirmation of data delivery. To address this issue the protocol incorporates the retransmission mechanism that ensures reliability at the data link layer [92].

4.5.2. Operation of the GOOSE protocol

When an event occurs in a substation automation system that uses GOOSE there is a change in the Status Number (Stnum) field of an IED indicating a failure. It is crucial to transmit this data to all devices subscribed to the GOOSE multicast group so that another IED can isolate the failure. To enhance the reliability in delivering GOOSE messages, the protocol employs the retransmission mechanism that relies on timing and relay messages to minimize the loss of data packets. This means that the same GOOSE message is sent multiple times with the interval between retransmissions increasing [93]. If the first GOOSE message carrying information gets lost, fast retransmission reduces the chances of losing that information since it’s repeated with the Stnum. The time it takes to achieve a

stable condition, through retransmission follows a geometric progression and typically takes around 1 second to ensure any electrical fault has been addressed by then.

4.5.3. GOOSE frame

Since GOOSE is operating directly on the IEEE 802.3 Ethernet frame, the frame structures are very similar. It contains the physical layer, the data link layer and the application layer. A graphical representation of the GOOSE message structure is shown in Figure 4.7.

Ethernet		802.1Q				Ethertype	GOOSE				
MAC dest.	MAC src	TPID	PCP	CFI	VID	Type	APPID	Length	Reserved 1	Reserved 2	goosePDU

Figure 4.7: Goose message format [93]

- In the Ethernet field, the source Media Access Control (MAC) address indicates the IED that sends the message, while the destination Multicast MAC address is the group that the message is being sent to.
- The 802.1Q is 4 bytes sized field responsible for adding a Virtual Local Area Network (VLAN) tag to the Ethernet frame, enabling the use of VLANs with varying priorities in messages.
- The Ethertype field, which is 2 bytes in size, denotes that this packet pertains to GOOSE and holds a value of 0x88B8.
- The Application Identifier (APPID) field is an attribute that identifies the application associated with a received GOOSE message.
- The length establishes the octet representing the total size of GOOSE Protocol Data Unit (goosePDU), along with 8 bytes of the APPID and Ethernet,

- The "Reserved" fields are designated for future use and have a value of 0 assigned to them.
- The goosePDU field is on the application layer and it contains the GOOSE application data such as Stnum, Sqnum, DatSet, AllData etc.

The goosePDU contains the actual data along with some useful metadata. In this thesis these data are characterized as the network features of the GOOSE frame. These features are further explained in Section 5.3.1.

4.5.4. Sampled Values (SV)

According to the IEC 61850 standard, messages related to sampled values are also time-critical. At the process level, the Sampled Values protocol is used to send analog values that are related to sensor or actuator measurements. The devices that are monitored at this level are mostly analog, so the data packets are formatted to represent analog values in digital format [94]. The traffic related to Sampled Values is continuous and provides protection functions and metering. Similarly, there is no acknowledgement feature in the sampled values protocol used to determine reception and interpretation of sample value packets. In this thesis, these values are broadly referred to as physical features, and they are further explained in Section 5.3.1.

4.6. Wireshark

Wireshark serves as a network protocol analyzer, utilized to capture and examine real-time GOOSE communication for substation networks based on IEC61850. It is typically used to analyze packets transmitted across Ethernet in any network [61]. Within the IEC-61850 standard, Ethernet-transmitted GOOSE messages can be scrutinized, and this software displays all the GOOSE related details. Furthermore, it can produce various

statistics for each message received via Ethernet, using the IEC61850 protocol. To visualize both the dataset and emulation capture, this research employs the Wireshark network analyzer software. Figure 4.8 displays the Wireshark interface of a captured GOOSE packet.

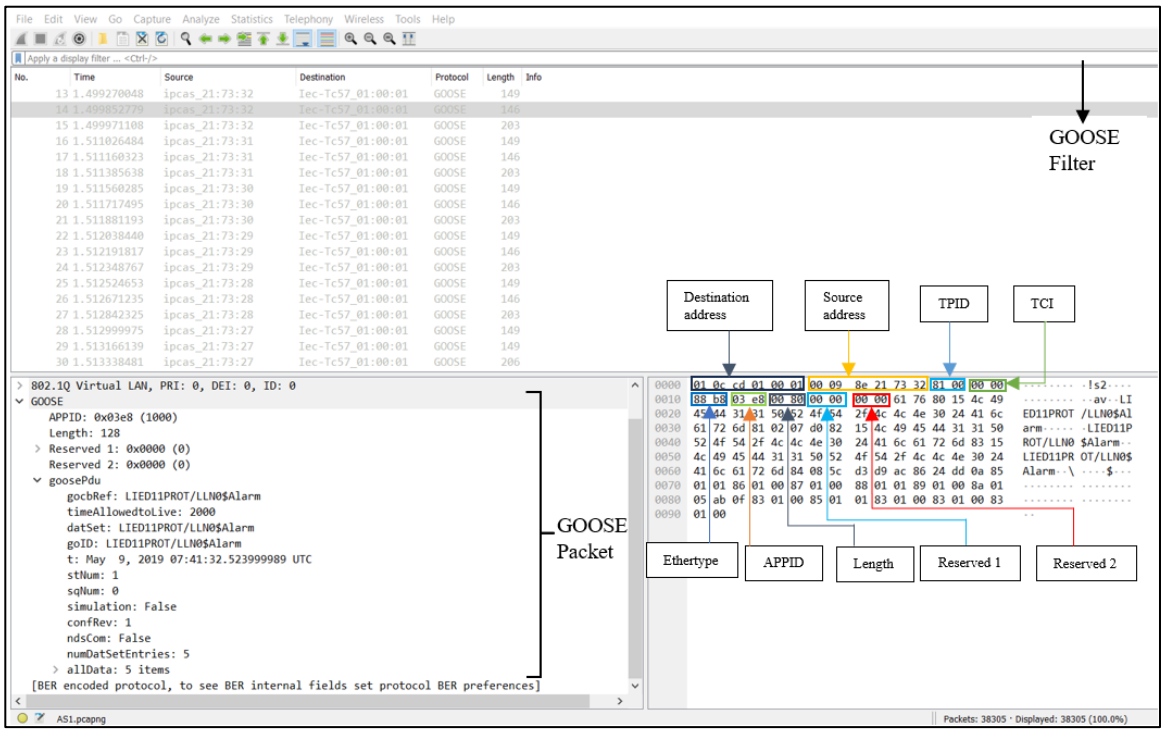


Figure 4.8: Wireshark user interface of the capture of a GOOSE packet.

4.7. Summary

This chapter provides an overview of the IEC 61850 standard, including its framework and implementation in the substation domain. It describes the information model used by devices that communicate with the IEC 61850-GOOSE protocol and its origin. The chapter also explains the concept of GOOSE interoperability, its naming convention, and the communication architecture used by the standard. In addition, the communication language used in IEDs, as well as how data are transmitted using

GOOSE and the message format in which GOOSE is structured are also discussed. Finally, the Wireshark software, which is utilized for analyzing the network and physical features of the GOOSE and SV messages received from the subscriber respectively, is discussed.

5. Model for the Detection and Classification of Cyberattacks in IEC 61850 Substation Automation Systems

5.1. Introduction

Several machine learning methodologies and their capabilities have been presented in Chapter 3. Various types of cyberattacks can be detected but if the attack is not classified it can affect the ability of Electrical Operators to implement targeted countermeasures which could have an impact on the reliability, stability, security, and quality of power supply to consumers[95]. The description of the cyberattacks studied will be presented. The machine learning model used in this work for the detection and classification of different cyberattack types will be introduced in this chapter. The model utilizes a bagging-based ensemble learning technique classifier, which consists of eager learners. This work presents the process of acquiring data, as well as preprocessing and feature engineering. The k-fold cross-validation method and number of learners parameter is used to optimize the performance of the proposed method. Moreover, an evaluation metric for the study of the obtained results will be discussed. Finally, this chapter also presents the essential physical and network features that are crucial for classifying different cyberattack types.

5.2. Attack Types Description

The most common cyberattack types that target substation automation systems are Data Manipulation (DM), Message Suppression (MS), and Denial of Service (DoS) [96]. Other prevalent attacks include FDIA attacks and replay attacks. In this chapter, a detailed description of all the aforementioned cyberattacks on IEC 61850 and the potential impacts are described.

5.2.1. FDIA Attack

False Data Injection Attacks aim to inject malicious measurements and modify the results. The FDIA could violate data integrity in various regions as transmission, communication, generation, control, etc. It can be seen in a different part of the smart grid that contains data. In this section, the FDIA will be evaluated in the grid without categorizing. It will be examined with the same approach for all regions. The main goal of FDIA attacks is to corrupt measurements and manipulate results. During transmission, communication, generation, and control, FDIA attacks can compromise the integrity of data thus jeopardising the cybersecurity. The attackers' objective is to manipulate the readings of multiple sensors, IEDs, and phasor measurement units (PMUs) with the intention of misleading the decision-making process of the smart grid [97]. In terms of cyberattacks, FDIA is considered as one of the most dangerous attacks. Presently, there is a significant attention on FDIAs as they are regarded as one of the most extensively studied cyber physical security attacks targeting smart grids [98] given the damage it has done on the systems that were affected by it.

Figure 5.1 illustrates the process of FDIA attacks in Smart Grid. Attackers have the ability to manipulate meter measurements by compromising meters on a local level, falsifying data packets that are sent to the control center through the exploitation of plaintext transmission protocol or altering the control center database [99]. Real-world incidents such as the December 2015 Ukrainian electric power blackout attack confirm these types of attacks [100]. Attackers can inject falsified monitoring data through compromising smart meters, sensors or IEDs, hijacking communication between sensor networks and the SCADA system, or intruding the SCADA system. As a result, the false

measurements can lead to an incorrect estimate of the smart grid state, which can mislead the control center to make wrong decisions and operations such as bad real-time electricity pricing and even large-area power failure accidents [101].

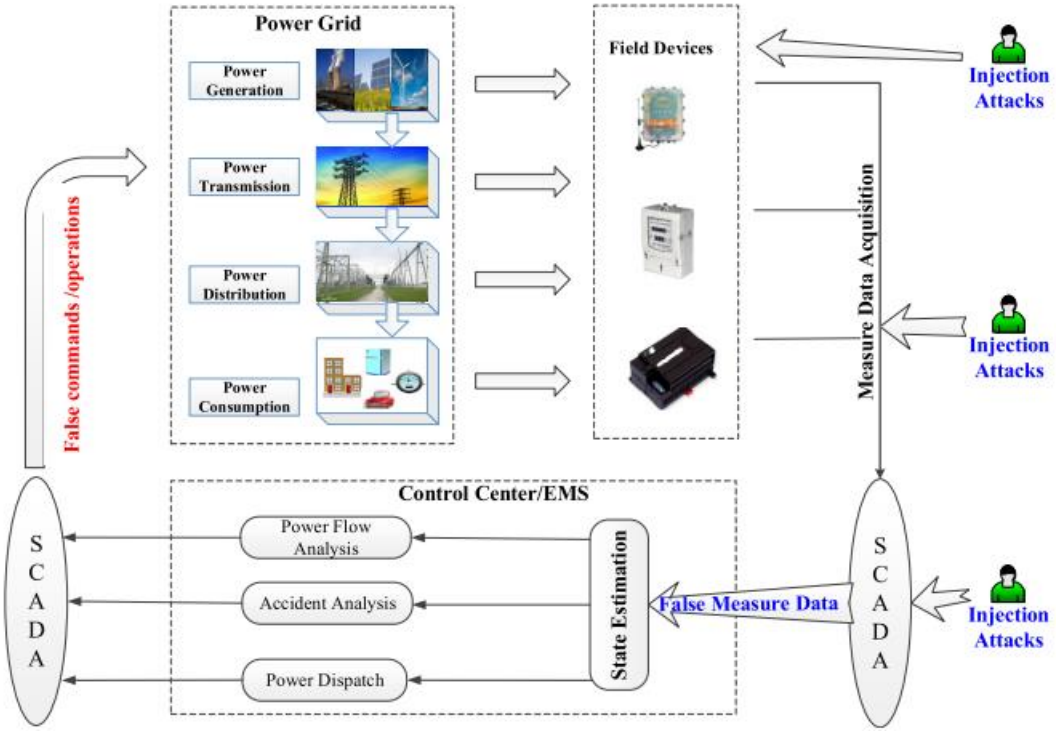


Figure 5.1: FDIA in smart Grid [37]

Steady state control, transient and auxiliary control, substation control and energy/load control are major operation and control blocks of power systems in which FDIA attacks can affect [102]. Figure 5.2 shows a detailed taxonomy of FDIA attacks against various power system control and operation blocks.

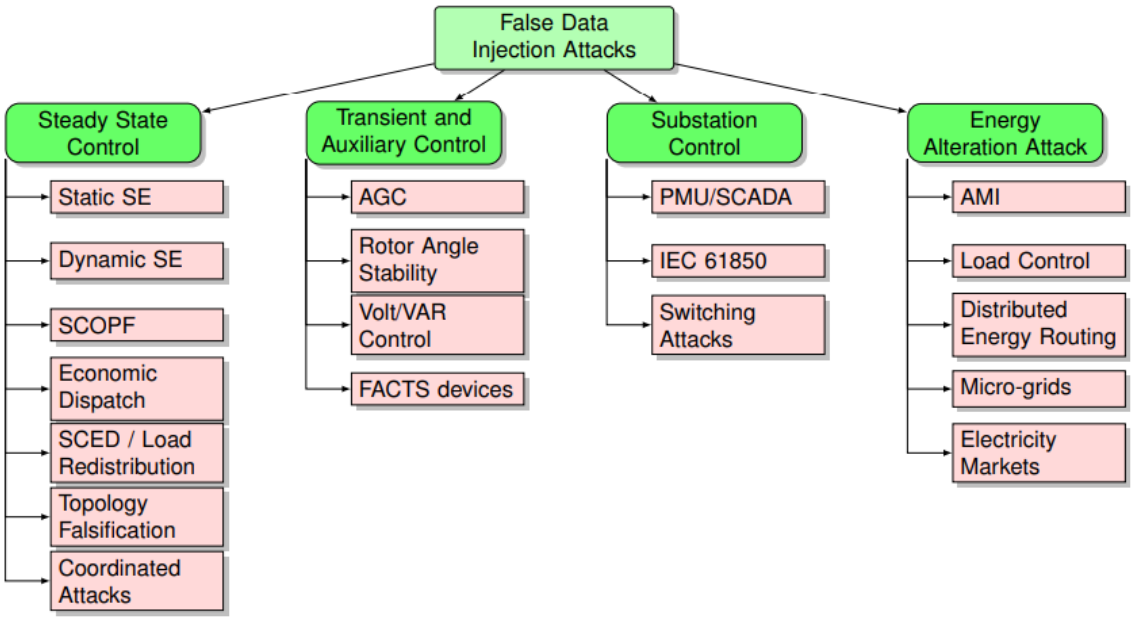


Figure 5.2: A taxonomy of FDIA attacks against various power system control and operation blocks [102]

There are several algorithms in power systems that are susceptible to FDIA attacks, including static state estimation (SSE), dynamic state estimation (DSE), optimal power flow (OPF), and security constrained economic dispatch (SCED). These types of attacks can involve a variety of tactics, such as redistributing load measurements, falsifying topology, or coordinating physical attacks by using false data. Additionally, not only steady state operations but also transient and auxiliary control blocks, such as rotor angle stability, automatic generation control (AGC), automatic voltage control (AVR), volt/VAR controls, and FACTS devices, are vulnerable to these attacks. Furthermore, substation control and communication architecture, including IEC 61850, PMU/SCADA data communication channels, are potential targets. The advanced metering infrastructure (AMI), residential

load controls, distributed energy routing algorithms, micro-grids, and electricity markets are also potential targets for data attacks.

5.2.2. Data Manipulation Attack

To tamper with measurements and manipulate data, attackers' resort to Data Manipulation (DM) attacks. These attacks exploit vulnerabilities by injecting manipulated network payloads into systems, aiming either at destabilizing the power grids or masking unauthorized alterations. The main goal of the attacker lies in compromising sensor and Intelligent Electronic Devices' (IEDs') readings. Their intended outcome is to deceive the substation automation systems decision-making process and by extension the entire smart grid [103].

In this type of cyberattack the contents or payload of network packets are modified in a way that goes unnoticed by both the publisher and subscriber [104]. The aim is to carry out a malicious act or an unauthorized action using the IEDs. According to [105] they are typically two forms of data manipulation attacks in IEC 61850 substation automation systems. The first case involves seizing the GOOSE control message packet and altering it with a message that allows the attacker to assume control and manipulate circuit breakers within a substation [106]. This attack can also be associated with Sample Value (SV) packets, where the attacker fabricates an analog value, which is then transmitted to a control center in the substation resulting in undesired operations. Through these attacks, the attacker gains control over IEDs and can cause unplanned power outages or even damage field devices within the substation [106].

In the second case, the attack type expands upon the earlier mentioned attack. However, this time it involves an automated approach using a malware script [107]. This malware

has the capability to capture, modify, and inject GOOSE message packets into the IEC61850 network. In order for the malware to carry out its objectives, it must first be installed on a computer within the substation network. This method of attack was successfully used against Kyivoblenergo, a regional electricity distribution company in Ukraine [108]. The researchers in [107] used this attack to exploit a weakness in GOOSE, where encryption and digital signatures are not feasible due to the requirement of immediate action within 4ms for any communication through a GOOSE message. Consequently, transmitted packets can easily be intercepted, modified, and retransmitted into the network without any form of encryption or digital signature. Figure 5.3 illustrates how this attack can be orchestrated.

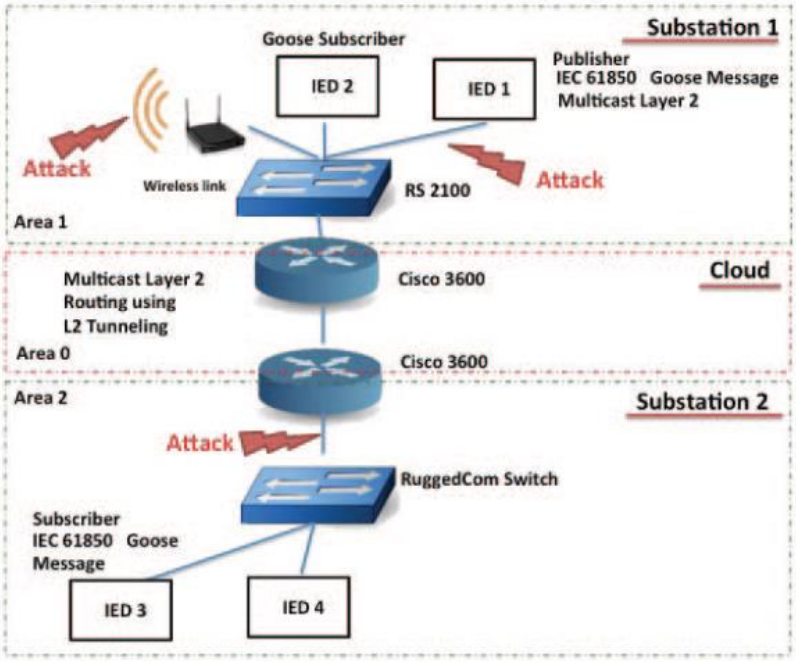


Figure 5.3: Network Diagram for Data Manipulation Malware Attack [107]

In carrying out the data manipulation attack, the verification of GOOSE messages in an IEC61850 network using Scapy (a Python program used for sniffing as well as packet

dissection, forging and sending back the network packets) is the first step. After the identification of GOOSE messages through Scapy, the GOOSE Ether-type, which is 0x88B8 is detected by capturing the raw packets. The next step is to decode the GOOSE message using the Abstract Syntax Notation One (ASN.1) defined in IEC 61850-8-1 protocol. After the decoding process, the malware script focuses on three fields: Status Number, Sequence Number, and Boolean values. These features are further explained in Section 5.3. The main aim of the malware script is to trip the circuit breaker by changing the Boolean value from true to false. However, for this attack to work, the values in the Status Number and Sequence Number fields must also be correct. Therefore, by examining the values of these fields in the GOOSE messages communicated between the publisher and the subscriber, the subscriber can then establish the accurate values to incorporate in the forged message.

5.2.3. Message Suppression Attacks

A Message suppression attack involves the unauthorized interception and alteration of protocol header fields in the GOOSE communication architecture, with the goal of obstructing the delivery of important messages or updates to legitimate IEDs in the network. In instances of message suppression attacks occurring within communication networks of GOOSE protocol frames, attackers can exploit vulnerabilities by manipulating sequence associated with these frames. By doing so, they are able to disrupt the subsequent arrivals of relevant information through these frames. Furthermore, the attackers can introduce modified versions of the GOOSE frames, which bear higher status numbers compared to previously transmitted ones. Subsequently, when subscribers process these

newly-introduced modified frames and encounter legitimate GOOSE frames with equivalent or lower status numbers, they will dismiss them as insignificant [107].

5.2.4. Denial-of-Service Attacks

A Denial of Service (DoS) attack is directed towards disrupting or disabling a service. Ultimately making it unavailable to users or leading to substantial delays. The intention behind such an attack is to overpower the systems' resources to render it inoperable. Substation systems utilizing Internet Protocol (IP) including GOOSE are frequently targeted by this form of assault [97].

A DoS attack has the potential to disrupt the proper functioning of an IED by preventing it from responding to genuine requests made by other IEDs. This can result into lack of power supply, unauthorized shutdown of substation equipment and other various devastating outcomes. The DoS attacks can be executed in several ways, one of which involves flooding the targeted IED with a substantial volume of GOOSE or SV messages until it becomes overwhelmed and renders the IED incapable of acknowledging valid requests [106]. Another strategy according to [109] entails carrying out a GOOSE poisoning attack, wherein the attacker aims to deceive the subscriber into accepting GOOSE messages with higher sequence numbers than those sent by the publisher. Therefore, only the attackers' GOOSE messages will be accepted and processed by the subscribers, rendering all the legitimate GOOSE messages from the publisher obsolete. An overview of the test bed for DoS attack by creating poisoned GOOSE attack is presented in Figure 5.4.

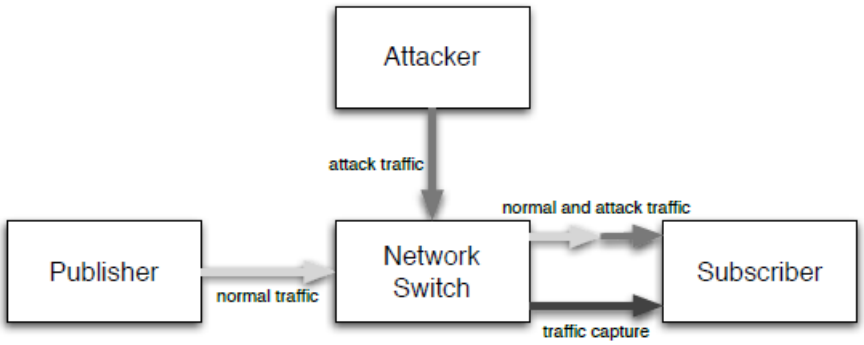


Figure 5.4: Test bed for DoS Attack

The work in [109] highlighted the three variations of the GOOSE poisoning attack. These variants are known as the high-status number attack, high-rate flooding attack, and semantic attack. In the high-status number attack the attacker sends a single spoofed GOOSE frame with an extremely high-status number from their source to a GOOSE subscriber, which is an IED. The goal here is to deceive the subscriber into accepting this spoofed frame as legitimate. In the high-rate flooding attack, the attacker takes a different approach by sending a series of spoofed GOOSE messages with increasing status numbers. Eventually, these spoofed frames surpass the expected status number on the GOOSE subscriber and create confusion in its functioning. The semantic attack involves two distinct phases. In the first phase the attacker carefully observes network traffic to gather information about the status numbers used in transmitted GOOSE messages and to identify patterns and rates of change. Armed with this knowledge, the attacker moves onto the second phase where they send spoofed GOOSE messages that have higher status numbers than what was observed in earlier transmissions [109].

A DoS attack targeting GOOSE messages could potentially disrupt the proper functioning of subscriber IEDs. As a consequence, only the GOOSE messages introduced

by the attacker would be processed by the subscriber IEDs. Persistent injection or flooding of these packets leads to a denial of service (DoS) attack preventing legitimate sender traffic from reaching the subscriber IEDs. Furthermore, this attack could allow the attacker to alter the payload of existing traffic.

5.2.5. Replay Attacks

During this type of attack, the attacker captures GOOSE messages and retains them for future use. Subsequently, the attacker retransmits the stored messages to initiate an action through the IED to the circuit breaker while it operates normally, to carry out undesirable outcomes. Also, in the case of SV message replay attack, the attacker can seize an SV packet containing precise power, current and voltage data and resend it numerous times to another protective device within the substation. This situation may result in unplanned outages if SV packets with identical power, current and voltage values circulate throughout the system multiple times [110].

5.3. Feature description

The messaging system used in the IEC 61850 protocol consists of network and physical features, which quickly sends out substation event details, including alerts and changes in status [92]. The following is a description of the network and physical features found in a SASs that utilizes the GOOSE protocol for communication.

5.3.1. Network Features

In GOOSE protocol, there are two identifiable categories of proprietary network features [43]. They consist of dynamic features, which are established through the analysis of the statistical trends related to volume and frequency of traffic [43]. The second category

is referred to as static features. These features play a significant role and are commonly selected through and extracted from different fields within a single GOOSE packet [45].

a. Status Number (Stnum)

The "Status Number" is a static feature that carries important information about the state of a particular device or process in the substation. This could involve the status of a circuit breaker (open or closed), a warning signal (Alarm), or other measurement information that needs to be communicated in real time within the substation network. The value of the Stnum is constant under normal operation and only changes when an event occurs.

b. Sequential Number (Sqnum)

The "Sequence Number" is a static feature that plays a pivotal role in maintaining the integrity and order of transmitted messages. Each GOOSE message in terms of circuit breaker status, alarm signal, measurement information is assigned a unique sequence number, which is incremented whenever a new message sent.

c. allData

The "allData" field is a static feature that carries a variety of data types, including Boolean control signal values (BCV). These signals represent binary state information, typically reflecting the status of a particular IED or process within the substation. Boolean control signals can indicate a range of conditions. In this thesis, they denote the control status of circuit breakers, earth switches and disconnectors. They also include protection alarm status of the protection systems within the substation.

d. numDatSetEntries

This static feature indicates the amount of data in the “allData” field. It is the number of elements that makes up the specific data set.

e. Source Internet Protocol Address

The Source Internet Protocol address or Source IP is a static feature that comprises of the unique identifier assigned to the originating device (such as a protective relay or a circuit breaker) that sends the GOOSE message.

f. Destination Internet Protocol Address

The Destination Internet Protocol (IP) address is a static feature that comprises of the unique identifier that indicates the intended recipient of the GOOSE message in the network.

g. GOOSE message heartbeat

A dynamic feature that refers to the mean time interval of the GOOSE arrival times.

h. GOOSE message length

This static feature describes the length of the GOOSE header.

i. GOOSE control block reference (gocbRef)

The GOOSE control block reference is a static feature that contains all the details of a pre-defined control block.

j. Application ID (APPID)

GOOSE application identification consists of information about the type of application sent from the publisher.

5.3.2. Physical Features

a. Current Measurement Values

The current measurement values serve as a vital physical feature that delivers real time sensor data regarding the electric current passing across different parts of the substation, namely transmission lines, transformers, and circuit breakers. These significant details are acquired and gathered by dedicated sensors known as current transformers (CTs) positioned within the IED. The CTs relay this information to monitoring and control systems, which allow for further examination of this collected data.

b. Voltage Measurement Values

Voltage measurement values allow for instantaneous assessment of electrical potential difference occurring at distinct locations across the substation. This valuable data is usually sourced from voltage transformers (VTs) or potential transformers (PTs) installed within the IED devices. Subsequently, this information is transmitted to monitoring and control systems found within substation facilities.

c. Active Power

Active power is a feature that holds utmost importance as it directly influences the successful completion of useful work required by an electrical load. To efficiently gather this power, Intelligent Electronic Devices (IEDs) are deployed throughout the substation infrastructure. These devices diligently observe voltage and current waveforms as part of their monitoring process.

d. Frequency

When discussing electrical systems, the term "frequency" refers to how many cycles occur within one second. This feature is measured in Hertz (Hz), and the numeric

value assumes great significance across North America's power system where an established standard frequency of 60 Hz holds firm. Considered as an essential parameter, this feature represents an equilibrium between electricity production and its subsequent consumption.

e. Circuit breaker Status

This feature describes the statuses of the different circuit breakers in the substation automation system.

5.4. Data Collection

This thesis makes use of datasets that contain information collected from IEDs operating on the IEC 61850 based GOOSE communication protocol in simulated distribution Substation Automation Systems (SAS). The synthesized datasets were collected from two different systems. The access to the network packet capture (PCAP) files for both systems is gotten from the public GitHub repositories of [59] and [111]. The Wireshark packet analyzer is used to analyze the files and gain a deeper understanding of the data.

The study in [59] presents an encompassing dataset comprised of network packets acquired from a test bed of five virtual machines (VMs). Within this setup, one VM emulates a 66/22kV primary plant while the remaining VMs simulate Intelligent Electronic Devices (IEDs). To explore potential vulnerabilities, two non-malicious behaviors termed normal and disturbance were implemented. Subsequently, a series of attack scenarios targeting the IEDs connected with IED1 and IED2 were conducted. The attacks were classified into two categories: Replay attacks and False Data Injection Attacks (FDIA),

wherein data within the primary message is modified or false data/messages are introduced. Lastly, these attack scenarios were executed under both normal operation and disturbance operation settings.

To further analyze the behavior of a system in various situations, the work in [111] constructed a dataset utilizing network packet data from a 66/11kv distribution substation featuring 18 IEDs. The dataset allowed them to simulate different scenarios including normal, disturbance and attack cases. During the normal scenarios they exposed the system to either variable or non-variable load conditions resulting in distinct current and power readings displayed by each IED due to changing energy demands over time. Conversely, the non-variable load scenarios exhibited steady energy flows as they experienced negligible variations in load demand. Disturbance cases were further divided into three categories: Busbar Protection, Under Frequency and Breaker Failure – each designed to test how the system responds to specific fault types. Lastly, four primary cases were defined for attack scenarios: Data Manipulation (DM), Denial of Service (DoS), Message Suppression (MS) and Composite Attack – aimed at creating a synthesized dataset for cybersecurity study in IEC 61850.

5.5. Data Preprocessing

After acquiring all the packet capture (PCAP) files from the two GitHub repositories - [112] and [113], a thorough preprocessing procedure was executed to prepare multiple datasets suitable for various experiments. This intricate process consisted of several stages: completion of any missing values, removal of redundant features, normalization of input data and encoding of labeled data. In depth analysis and clarification on each step undertaken during the preprocessing process is presented below.

1. Missing value Imputations

One of the most common challenges when working with simulated datasets that attempts to mimic real world dataset is managing missing values. These missing values can appear as NaN values, blanks, or other placeholders and can negatively affect the evaluation of machine learning models if the models were trained on datasets with missing values. To address this issue, several techniques have been proposed in recent years. These include hot and cold deck imputations, mean imputations, and extrapolation and interpolation imputations, as well as zero imputation. Hot deck and cold deck methods involves filling in missing data with values from similar records (hot deck) or a donor pool (cold deck). Mean imputation replaces missing values with the mean of available cases. Extrapolation and interpolation estimations assume specific trends in the data to estimate missing values. In this study, the zero-imputation technique is utilized to handle missing data. This technique involves replacing missing values for a particular feature with a fixed value of zero. This technique is selected because it aligns well with the nature of the data collected and the purpose of our analysis. The dataset used in this thesis included instances where a Denial of Service (DoS) attack was simulated. Consequently, a significant portion of the data had blank spaces instead of physical measurement values, indicating that no data was recorded for those instances where packets were parsed without physical value information. These blank spaces were attributed to DoS attacks, emphasizing the importance of effectively handling missing values in datasets that have been impacted by cyber-attacks.

2. Feature Selection

Feature selection techniques play an important role in selecting the most important features to decrease training time and to reduce model overfitting thus increasing the algorithm's accuracy [66]. Feature/Predictor Importance was the selection method used. Predictor Importance (PI) is a method that generates a score for each of the input features for a particular model. A higher score indicates that the feature will have a greater impact on the model used in classification. This method is highlighted in the classification process of the proposed approach in Section 5.6.7. In this thesis, the feature dimensions were reduced to relevant features. Using the PI method, features with high scores were selected. These features include GOOSE Heartbeat, Goose length, Status number, Sequence number, allData and all the physical features. Also, after preprocessing, some features were derived from the selected features, as shown in Table 5.1, Table 5.2, Table 5.3 Table 5.4. Conversely, removed features include gocbRef, APPID, Source IP and Destination IP. These features were removed because through the PI method, these features had very low scores therefore signifying that they had very little to no impact on the training model. In addition, having these types of features can make the model learn based on irrelevant features, hence decreasing the accuracy of the model. Attributes, including some network features and sample values, can have a significant impact on the detection accuracy of malicious traffic in the network.

3. Data Normalization

The accuracy of identifying malicious traffic in a network can be significantly affected by the choice of input features. With this in mind, carrying out data normalization on the datasets is imperative as machine learning algorithms face the challenge of recognizing

features that have varying scales. Data normalization is necessary to organize input data and to ensure consistency and similarity across all fields and records. By normalizing or standardizing the data, potential biases can be mitigated, leading to more accurate analysis and predictions.

From the 4-IED dataset, six physical features related to the current were extracted and processed. These features are based on the 66/22kV primary plant architecture as shown in Figure 5.5. Each of the six physical features is the average value of one horizontal level within the substation. Specifically, these features represent the average currents in the 66kV lines and circuit breakers (high voltage), transformer's 66kV side windings (w1), transformer's 22kV side windings (w2), transformer circuit breakers, 22kV lines and circuit breakers (low voltage), and all the feeders.

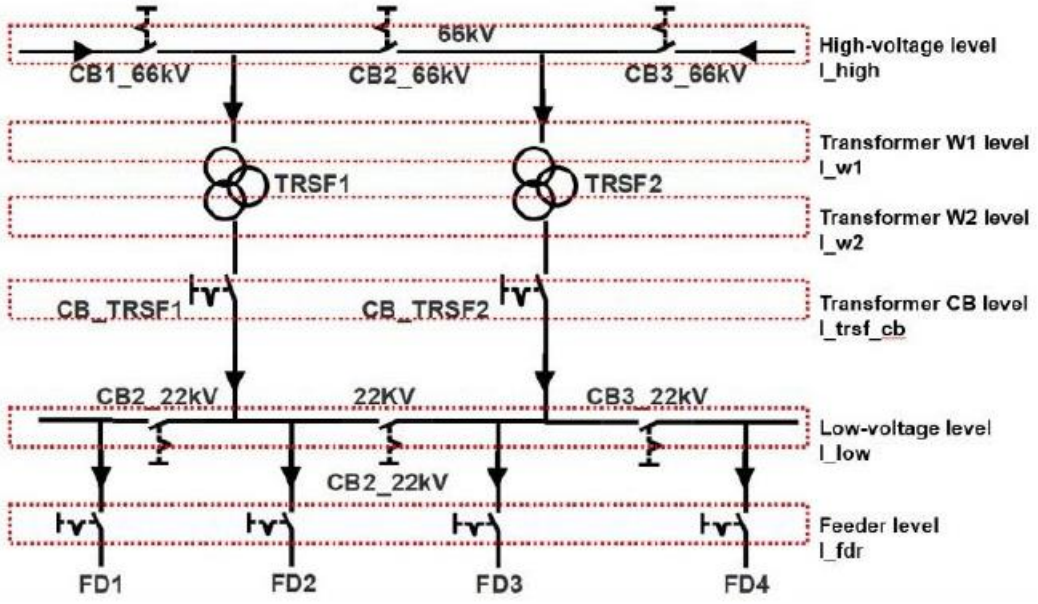


Figure 5.5: Summary of the Physical Features of the Primary Plant, Categorized by their Horizontal Levels.

This configuration will assist in figuring out the IED that has been compromised whenever there is a presence of an attack due to inconsistent readings from any of the six horizontal levels. Also, all circuit breaker statuses on the high voltage side, transformers and low voltage side were summarized into one feature. The circuit breaker statuses are compiled into a specific sequence and are represented as a binary number, given that the circuit breakers only have an open state (0) and a closed state (1). In this thesis, the processed network features, Difference in Stnum "Dif_Stnum" and Difference in Sqnum "Dif_Sqnum" is used to determine any changes in the "Stnum" and "Sqnum" values compared to their previous values. This helps to maintain a constant value of 0 for "Stnum" and 1 for "Sqnum". If there is a deviation from these values, it may indicate a potential attack. Also, the feature "Dec_allData" involves converting the Boolean control signals of "allData" field extracted from wireshark from binary to decimal format. For instance, the Boolean control value [1, 1, 1, 0] present in the "Dec-allData" field is converted into a binary number "1110," which was then converted to the corresponding decimal number "14". Table 5.1 and Table 5.2 list the processed network and physical features utilized.

Table 5.1: 4-IED Dataset Network Features Description

Network features	Description
GOOSE Heartbeat	The temporal duration between two adjacent packets
GOOSE length	The length of the GOOSE header
Dif-Stnum	The differential value between the current Stnum and the previous Stnum
Dif-Sqnum	The differential value between the current Sqnum and the previous Sqnum
numDatSetEntries	The amount of data in the “allData” field
Dec-allData	The decimal number of converting all Boolean values in the “allData”field

Table 5.2: 4-IED Dataset Physical Features Description

Physical features	Description
I-high	The average current values among high-voltage level
I-w1	The average current value among transformers' winding 1
I-w2	The average current value among transformers' winding 2
I-trsf-cb	The average current value among all transformers' circuit breakers
I-low	The average current value among low voltage level
I-fdr	The average current among all feeders
Bin-cb-status	The binary sequence of statuses of all circuit breakers

The standardization of the network and physical features contained within the 18-IED dataset also helps mitigating potential biases as well due to the diversified nature of such dataset. All the Boolean control values (BCV) were transformed into unique numerical values to reduce the computational costs. The control feature "Dec" represents the Boolean to decimal conversion of the following Boolean control values: Circuit breaker Open/Close Status (1), Disconnector Open/Close Status (1), Protection tripped (0), and Intertrip command send (0), which correspond to the decimal number "12". Also, linear transformation was performed on the other input data features such Stnum, voltage and power using the Min-Max normalization technique. This is used to scale the values of the features to a range between 0 and 1. The minimum and maximum values are obtained from the data, and each individual value is then substituted according to equation 5.1.

$$v_i` = \frac{v_i - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A \quad (5.1)$$

Where, A represents Stnum, voltage and power features, $\max_A$ and min_A denote the maximum and minimum absolute values of the attribute data A respectively. The term $v_i`$ refers to the newly computed value for each packet, while v_i represents the original value of each packet. The new_max_A and new_min_A refer to the maximum and minimum values of the desired range, that is, the boundary values of the features normal operating range, respectively. Table 5.3 and Table 5.4 lists the processed network and physical features used in this work and their description.

Table 5.3:Network Features Description of the 18-IED Dataset

Network features	Description
BCV_Dec	Boolean to decimal conversion of Information pertaining to the control status
v`-Stnum (measurements)	The Min-Max normalized Stnum values within the measuremer reading of an IED
v`-Stnum (Status)	The Min-Max normalized Stnum values from the Status measurement of an IED
v`-Stnum (Alarm)	The Min-Max normalized Stnum values from the Alarm updates of an IED
Dif-Sqnum (measurements)	The difference observed between successive Sqnum values within the measurement reading of an IED
Dif-Sqnum (Status)	The difference observed between successive Sqnum values from the Status values of an IED
Dif-Stnum (Alarm)	The difference observed between successive Sqnum values from the Alarm updates of an IED
Numofblank	The quantity of absent attributes within a singular data entity.
v`_Stnum + v`_Sqnum < 0:	Summation of v`_Stnum and v`_Sqnum. Results less than 0 gives a flag (1111).

Table 5.4: Physical Features Description of the 18-IED Dataset

Physical features	Description
v`- Current Line 'L1’	The difference observed between two successive current readings in Line 1
v`- Current Line 'L2’	The difference observed between two successive current readings in Line 2
v`- Current Line 'L3’	The difference observed between two successive current readings in Line 3
v`-Voltage Phase 'L1-N’	The Min-Max normalized line 1 to neutral voltage readings
v`-Voltage Phase 'L2-N’	The Min-Max normalized line 2 to neutral voltage readings
v`-Voltage Phase 'L3-N’	The Min-Max normalized line 3 to neutral voltage readings
v`- Active Power	The Min-Max normalized active power readings
50-Frequency	The deduction of the prevailing frequency from the designated nominal frequency

4. Encoding Labeled Data

A crucial element of data preprocessing involves the encoding of labeled data. In this thesis, all disturbance operations were labeled as ‘Disturbance,’ while normal traffic was labeled as ‘Normal.’ For the 4-IED dataset the attack types described as False Data Injection Attack and Replay attack were labelled as FDIA and Replay respectively. The attack types in the 18-IED dataset described as Data manipulation attacks, Message suppression attacks, and Denial of service attacks were labeled as DM, MS, and DoS, respectively.

5.6. Fine Tree-based Bagging Ensemble (FTBE) Approach for Cyberattack Classification

This section describes the proposed approach for classifying different types of cyberattacks using a bagging-based ensemble learning technique also known as bootstrap aggregating. Previous research has shown that decision trees used in recursive partitioning exhibit instability when working with small datasets [114]. Such instability can have adverse effects on both classification accuracy and tree structure. To mitigate these issues, a strategy involving multiple classifiers can be adopted [115]. The method of constructing an ensemble involves using distinct subsets of training data along with a decision tree serving as the base learner. Decision tree classifiers are generally classified into three categories: Coarse Tree, Medium Tree, and Fine Tree. In this work, Fine Tree (FT) classifier type is selected based on its superiority over other classifier types [116]. The explanation of the FT model’s implementation within the bagging-based ensemble learning approach is presented. The detailed information regarding the parameter settings necessary for achieving optimal accuracy is also provided in this section.

5.6.1. Fine Tree Model

The classification of decision trees can be considered unstable due to the fact that even minor changes in the training data can have a significant impact on the overall structure of the tree [117]. To manage the complexity of the tree, various stopping criteria are utilized [118]. These criteria are typically assessed using metrics like the total number of nodes and leaves, the depth of the tree and the number of attributes used. During the growth process of the tree, it will keep expanding until a stopping criterion is satisfied. These criteria may involve conditions such as:

- The tree's depth has reached its maximum limit.
- The ideal splitting criteria does not exceed a particular threshold.
- If the node were to be split, there would be insufficient cases in one or more of the child nodes as it would fall below the minimum number of cases required for child nodes.
- The number of cases in the terminal node is lower than the minimum number of cases required for parent nodes.

Over fitting of data can occur when the decision-making process becomes overly reliant on irrelevant features [118]. To address this issue in traditional decision tree learning, researchers have proposed different solutions. One such solution is pruning as demonstrated in the work presented in [119]. Their method, known as depth impurity (DI) pruning, takes into account the complexity of sub-trees and preserves those sub-trees that generate relevant decision rules. However, it was discovered that this method did not improve the classification efficiency. In a subsequent study conducted by [120], various pruning algorithms for estimation trees were analyzed to determine the most suitable one

for specific situations. It was deduced that depth control of the tree and proper feature selection can greatly enhance the performance of an algorithm.

In this thesis, controlling the depth of the model was used to prevent overfitting. In doing this three decision tree classifier types are considered – Coarse Tree, Medium Tree, and Fine Tree. Coarse tree is a type of decision tree that consists of few leaves to make coarse distinctions between classes with a maximum number of 4 splits. The Medium tree is a type of decision tree with relatively more splits, up to 20 maximum number of splits. The Fine tree model is a type of decision tree that consists of many leaves to enable many fine distinctions between classes. However, the maximum number of splits is set to 100 in order to control the depth of the model to prevent overfitting. The process when the fine tree model was compared with the medium and coarse tree models, it performed better as it was more adaptable to the datasets it was not initially trained on. The work in [120] used 20 machine learning algorithms to compare the performance of two cross validation techniques on the University of California, Irvine (UCI) datasets. Coarse tree, medium tree and fine tree were amongst the 20 algorithms. Overall, from the results, the Fine tree algorithm performed very well and also had a higher accuracy than the other two algorithms for both cross validation techniques. The fine tree classifier has proven to be one of the most efficient techniques [116].

5.6.2. Finding best split

In growing the individual fine tree models, the input data are first preprocessed and then the outcome of this process is a set of physical and network features. The selected physical and network features serve as the input to the classification tree generation process. Initially, the numeric values of the continuous attributes are sorted in ascending

order. Next, the Candidate Split Positions (CSP) are identified by taking the midpoint between two adjacent values for each feature, in order to evaluate the Gini index of a candidate split position.

$$CSP = \frac{sorted\ value_2 - sorted\ value_1}{2} \tag{5.2}$$

The Gini index is then calculated using:

$$Gini(t) = 1 - \sum_{i=0}^{c-1} [p(i|t)]^2 \tag{5.3}$$

Where $p(i|t)$ is the portion of observation that belongs to class i at a given CSP node t and c, the number of class labels. Then, the Weighted Gini index $\left(Gini_{weight}\right)$ is computed as:

$$Gini_{weight} = \sum_{t=0}^{n} \frac{X_t}{T} \times Gini(t) \tag{5.4}$$

Where, X_t is the number of scenarios in node t, T is total number of scenarios, $Gini(t)$ is the Gini index value at a given node t and n is the number of nodes. The same process is repeated for the remaining CSP. The CSP that produces the lowest Gini weighted average is determined to be the best split point. This process is repeated among all the features and is then used in the construction of the fine trees.

5.6.3. K-fold Cross Validation

Datasets generated from distribution substations are often imbalanced. There are cases were attack scenarios may be few compared to the other instances. In order to achieve a classification model approach that handles imbalanced dataset, thorough training and

evaluation of each instance present in the dataset must be carried out. In turn averaging all recorded accuracies across the trained instances. To avoid cases like underfitting or overfitting for specific trained datasets, k-fold cross validation serves as a method for developing a well generalized model.

The k-fold validation method divides the data into “k” segments or folds of equal or nearly equal size. In each iteration, the model is trained and is tested on these folds. During each iteration one-fold is kept aside for testing while the model is trained on the remaining “k-1” folds as shown in Figure 5.6. To determine the overall accuracy of the model, the accuracy achieved in each iteration is averaged. Additionally for each fold an error value is calculated using equation (5.6). The total error (ε) is then obtained by summing up the errors from all *k* iterations.

$$\varepsilon = \frac{1}{k}\sum_{i=1}^{k} \varepsilon_i \tag{5.5}$$

The average accuracy of the k-fold accuracies (α) determines the performance of the classification model [121].

$$\alpha = 1 - \varepsilon \tag{5.6}$$

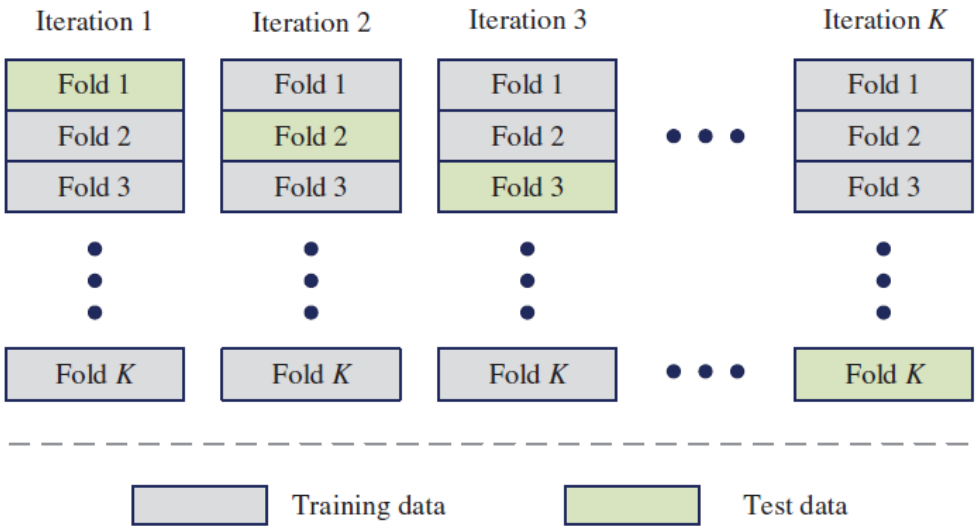


Figure 5.6: Process of the K-fold cross validation method [122]

5.6.4. Bagging-based Ensemble Classifier

Bagging creates an ensemble of classifiers by sampling with replacement from the set of training data to create new training sets called “bags” [123]. Bagging-based ensembles train their base learners independently from each other, and they use data transformations to promote diversity into the predictions of the model. When a base learner, typically a weak learner, is considered individually on large imbalanced datasets, they sometimes provide inadequate prediction accuracy [124]. This limitation can be addressed by combining multiple models into one that delivers better overall performance. In this work, a weak learner – Fine Tree, serves as the base learner. A weak learner is an appropriate description for this type of base learner due to its tendency to exhibit high bias or high variance after computation [124]. A model with high bias implies that it has not thoroughly understood the underlying data. This issue is independent of the data distribution and can result in future predictions that are unrelated and incorrect. Conversely, when a model overlearns from the data (high variance), accurate prediction of subsequent points becomes

challenging due to large variations between individual data points. Figure 5.7 visually depicts the concept of high bias and high variance in a weak learner.

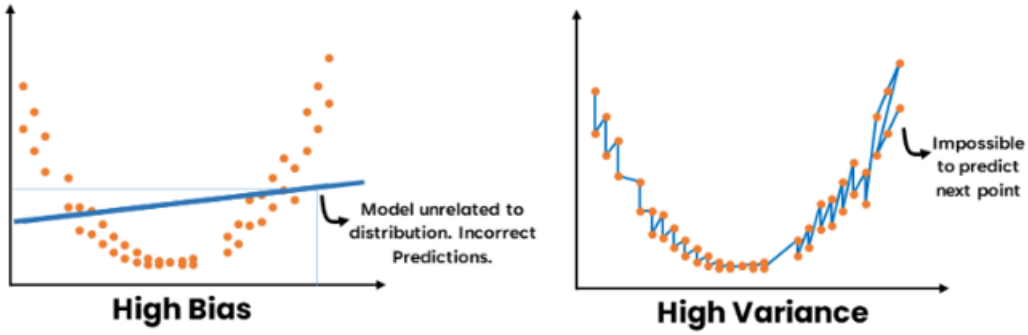


Figure 5.7: High Bias and High Variance Performance of a Weak Learner [124]

Weak learners, which can be characterized as models that either have a high bias (underfit) or a high variance (overfit), face challenges when it comes to properly generalizing and predicting accurately when in isolation. Ensuring a balance between bias and variance is crucial in order to develop a model that can accurately generalize from the data it was trained on to new unseen data. Ensemble learning serves as a strategy that aims to achieve this balance [125]. Depending on the specific model being used, Ensemble learning techniques are able to address either high bias or high variance in weak learners, resulting in a more well rounded and robust learner. As a result the model becomes more generalized and is able to provide accurate predictions as shown in Figure 5.8.

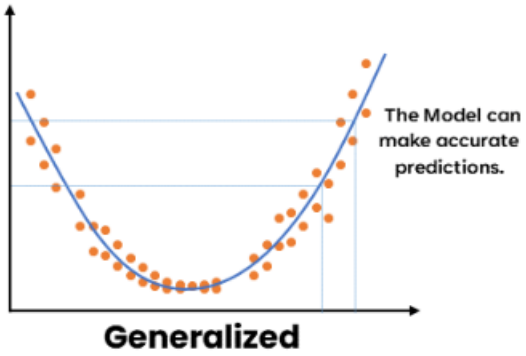


Figure 5.8: Plot Performance of an Ensemble Learner

5.6.5. Fine Tree Bagging-Based Ensemble (FTBE) Approach

After the construction of the fine trees, the bagging-based ensemble learning approach creates several randomized fine trees by determining the best split among the dataset's attributes as explained in Section 5.6.1, which is done by assessing the impurity of the tree nodes using the Gini index. From the initial training dataset containing x-number of instances, y-number of subsets of data are created from the training set. The y in this case is referred to as the Number of Learners. A subset of X sample points is taken from the initial dataset for each subset. Each subset is taken with replacement, which means that a specific data point can be sampled more than once. For each subset of data, the corresponding fine tree is trained independently by and evaluated on every instance in the dataset. This was done using the K-fold cross validation method as explained in Section 5.6.3. The models created from this process are homogeneous, meaning that they are of the same type. The ensemble E, which comprises m decision tree classifiers, is expressed as:

$$E = \{FT_1, FT_2, FT_3, \dots, FT_m\} \tag{5.7}$$

Once the individual trees are generated from the randomized subsets, the resulting classification outcomes are determined through a majority voting scheme. Each of the

classifiers, FT_i, makes a prediction y_i for each data point in the test set, where y_i represents the predicted class out of the k possible classes $(y_1, y_2, \dots, y_k)$. To determine the final predicted class for a given data point 'x', the mode of the classes predicted by the fine tree classifiers in the ensemble is calculated using:

$$y_m(x) = mode\{FT_1(x), FT_2(x), FT_3(x), \dots, FT_m(x)\} \quad (5.8)$$

Where, $FT_i(x)$ denotes the prediction $y_i(x)$for a given data point 'x' and it represents the selection of the most commonly predicted class among the fine tree classifiers. Figure 5.9 illustrates the approach used to grow the fine tree classifiers for feature extraction and implementation.

5.6.6. Number of Learners

When developing the ensemble classifier model, it is important to find the number of fine trees required for optimal outcomes. This directly correlates with the number of learners employed during the process. By increasing the number of learners, a greater subdivision of data can be achieved, enabling each fine tree to undergo more comprehensive training. A very large number of learners however, can increase the complexity of the model. Therefore, finding the optimal number of learners that makes the ensemble learning model inexpensive is imperative.

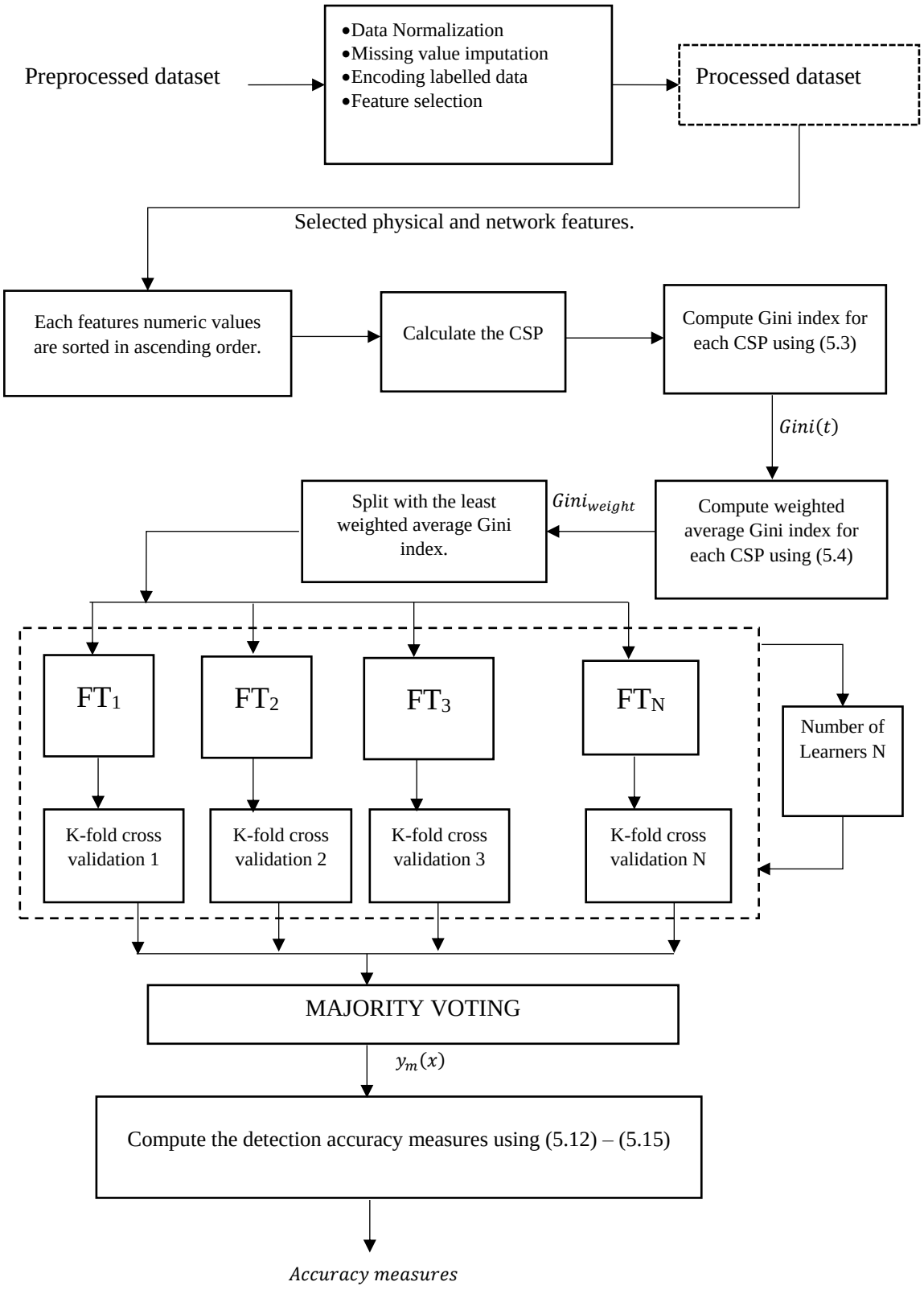


Figure 5.9: Feature Extraction and Class Prediction of the Proposed Approach

5.6.7. Predictor Importance (PI)

An advantage of this approach is the use of Predictor Importance (PI), which helps identify the most significant features for each fine tree classifier. This function assesses a features importance after the model is trained [126]. To calculate this metric, the changes in node risk caused by splits on every predictor are summed, and then divided by the total number of branch nodes [127]. This enables the determination of which features have the most impact on the classification outcomes of the fine tree.

$$PI_i = \frac{\Delta R_i}{N_{branch}} \tag{5.9}$$

Where, R_i is the node risk of the node i, and N_{branch} is the total number of branch nodes. The $\Delta R_i = R_p - R_{Tc}$ is the difference between the node risk of the parent node and the total node risk of the children's nodes. A node risk is stated as a node impurity weighted by the node probability:

$$R_p = P_p E_p \tag{5.10}$$

The total risk of the children's node is calculated using:

$$R_{Tc} = P_{c1}E_{c1} + P_{c2}E_{c2} + \cdots + P_{cn}E_{cn} \tag{5.11}$$

Where, P_p is the node probability of the parent node, and E_p is the node impurity of parent node, which is obtained using the Gini Index from equation (5.3) . P_{cn} and E_{cn} refers to the node probability and node impurity of the 'n' number of children nodes respectively.

5.7. Evaluation Metrics

Assessing a classifier's efficacy requires evaluating its performance using evaluation metrics. In this thesis, various measures such as accuracy, precision, recall, and F1-score are employed to evaluate the proposed approach for the classification of cyberattack types. The significance of this stage cannot be overstated as it determines overall effectiveness of the classifier used.

5.7.1. Accuracy Measure

The classification accuracy measures the total number of correctly predicted cases in relation to the overall number of cases.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.12}$$

The instances denoted by *TP*, *TN*, *FP*, and *FN* refer to the number of True Positive, True Negative, False Positive, and False Negative rates, respectively. In *TP* cases, the attack type is classified accurately. For *FP* cases, a normal or a disturbance instance is misclassified as an attack type. In *FN* cases, an attack scenario is mistakenly classified as a normal or disturbance scenario. Finally, for *TN* cases, normal and disturbance scenarios are classified accurately.

5.7.2. Precision

Precision is a measure of how well attack predictions by the classifier match actual attack instances. It determines the fraction of accurately predicted attacks. A classifier's false positive rate decreases as its precision increases [121].

$$Precision = \frac{TP}{TP + FP} \tag{5.13}$$

5.7.3. Recall

The recall metric calculates the proportion of attack instances that were accurately identified as attacks by the classifier. This metric indicates the effectiveness of the classifier in identifying attacks. A classifier with a high recall score will have a minimal number of attack scenarios that are inaccurately classified [121].

$$Recall = \frac{TP}{TP + FN} \tag{5.14}$$

5.7.4. F1-score

The F1-score is the metric that calculates the harmonic mean of precision and recall [121].

$$F1 - score = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \tag{5.15}$$

5.7.5. Confusion Matrix

The confusion matrix is a representation that helps evaluate the accuracy of a classification model. It presents a table, as shown in Table 5.5 that displays the number of correctly and incorrectly predicted records. The table shows predicted classes in columns and true classes in rows. It shows nine instances where TP_N represents cases classified as Normal by the classifier, and they were actually normal cases. However, E_{ND} is a sample

from the normal class that was wrongly classified as a disturbance. Therefore, the sum of E_{ND} and E_{NA} ($FN_N = E_{ND} + E_{NA}$) represents false negatives in the Normal class, indicating all normal cases that were misclassified as disturbance or attack cases. On the other hand, the sum of E_{DN} and E_{AN} ($FP_N = E_{DN} + E_{AN}$) represents false positives in the Normal class, indicating all non-normal cases that were misclassified as normal cases.

Table 5.5: Confusion matrix

True Class \ Predicted Class	Normal	Disturbance	Attack type
Normal	TP_N	E_{DN}	E_{AN}
Disturbance	E_{ND}	TP_D	E_{AD}
Attack type	E_{NA}	E_{DA}	TP_A

5.8. Summary

This chapter describes the methodology used in the detection and classification of cyberattacks in IEC 61850 substation Automation Systems. Firstly, the cyberattacks types and features of the GOOSE communication packets are described. The proposed approach is developed by first preprocessing the dataset through data normalization, missing value imputation, encoding labelled data and followed by feature selection, which utilizes the prediction importance technique to select features that impacts the model's classification. Furthermore, the Fine Tree Bagging-based Ensemble (FTBE) learner is introduced where the growing of the Fine Tree classifier and the implementation of the bagging-based ensemble is described. In addition, the k-fold cross validation method and number of learners hyper parameters for detection and classification of the cyberattacks are described.

Finally, the FTBE approach is developed along with the description of the evaluation metrics used assess the model.

6. Results and Evaluation

6.1. Introduction

This chapter assesses the performance of the proposed FTBE approach developed in this thesis and compares to the methodologies highlighted in Chapter 3. Two test systems that employ IEC 61850 GOOSE communication were utilised to generate datasets pertaining to the cyberattack types investigated in this study. The [59] and [111] public GitHub repositories contain the network packet capture (pcap) files for the respective systems. First, the dataset is preprocessed through data normalization, missing value imputation, encoding labelled data and feature selection. The fine tree then takes into account the selected features through the PI technique described in chapter 5 to find the best split. Next, the k-fold and number of learners hyperparameters of the bagging ensemble learner were tuned so that the classification accuracy could be determined. In order to evaluate the sensitivity of the proposed approach to the choice of the training and the testing dataset, different cases representing different combinations of line IEDs (LIEDs) for both systems are used in this work. Furthermore, the sensitivity of the proposed approach to the variations in the k-fold parameter and the number of learners is also carried out on the results to achieve the optimal settings for k-fold and number of learners.

6.2. Test System Description

In order to test the proposed approach, a dataset is necessary for training and testing. The systems used in this study replicate not only the physical system of typical distribution level substations but also a number of the important electrical protection operating scenarios under a variety of disruptions, which are then followed by a number of potential cyber-attack scenarios.

6.2.1. Test System 1

The test system created by [59] simulated five virtual machines (VM) that ran on Oracle VirtualBox. One of the VM simulates a 66/22kV distribution substation automation system (Primary plant) using MATLAB/Simulink. The system consists of a 66kV high voltage line, two transformers, a 22kV low voltage line, four feeders, and nine circuit breakers. The other four VMs represents four Intelligent Electronic Devices (IEDs) simulated using OpenPLC. The IEDs are made up of three instantaneous overcurrent protection devices (IED_PIOC) and one circuit breaker failure protection (IED_BFP). These IEDs are situated at transformer 1(IED_PIOC_TRSF1), transformer 2 (IED_PIOC_TRSF2), 22 kV circuit breaker 2 (IED_BFP), and on the feeder side (IED_PIOC_FDR), as shown in Figure 6.1. Ten short-circuit fault blocks were set up to generate events at ten different locations.

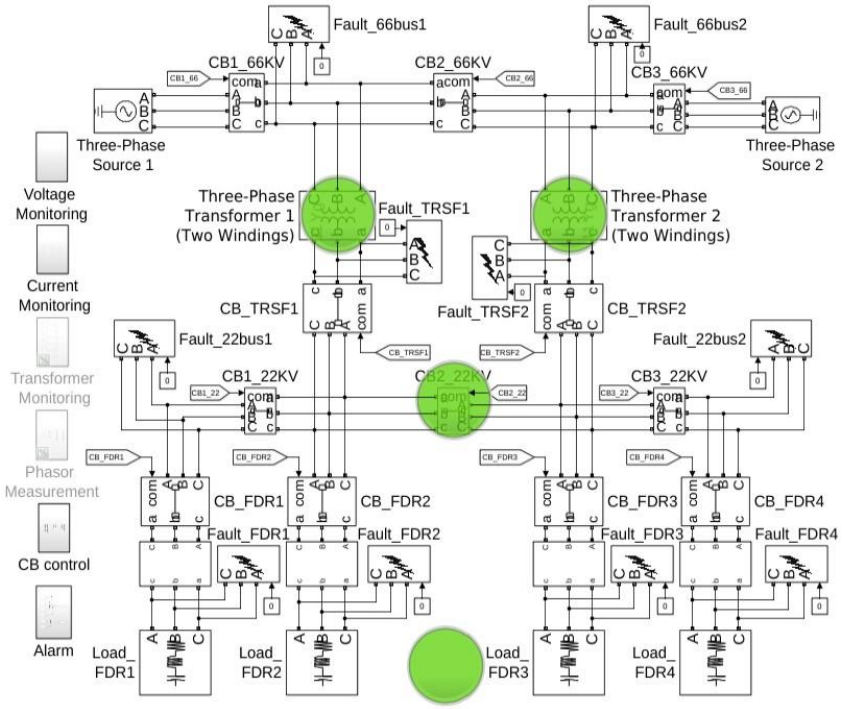


Figure 6.1: 66/22kV Substation Test System used to Generate the 4-IED Dataset.

C/C++ is used to create VM communication networks such as GOOSE trip messages between IEDs and the primary plant, according to the IEC 61850 library (libiec61850). Each VM also has OpenPLC, MATLAB/Simulink, and "libiec61850" interface programs. As illustrated in Figure 6.2, the interface program in VM-IEDs reads analogue values from Simulink in VM-Primary Plant through User Datagram Protocol (UDP) packets and passes them to OpenPLC. The program reads OpenPLC digital signals and delivers them to "libiec61850" to construct GOOSE packets. After VM-Primary Plant's "libiec61850" program receives GOOSE packets, the interface program reads digital signals from decoded packets and sends them to Simulink through UDP packets. The IEDs send GOOSE messages to primary plant circuit breakers through the central process bus.

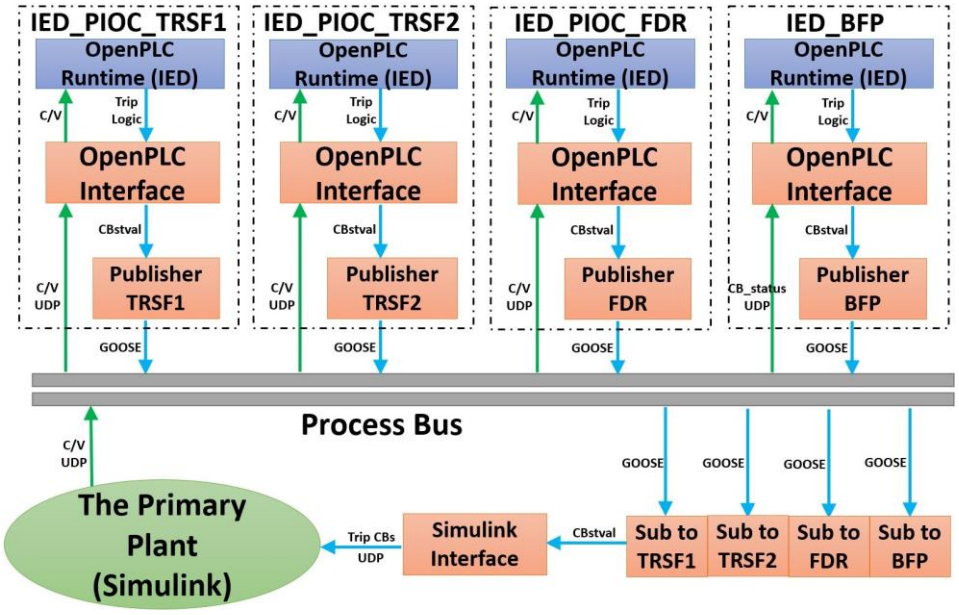


Figure 6.2: Structure of Communication Networks in the Test system

6.2.2. Test System 2

The test system by [111] simulated a 66/11kV substation automation system. This substation consists of 18 Intelligent Electronic Devices (IEDs); 2 Transformer IEDs (TIEDs), 14 Line Feeder IEDs (LIEDs), 1 Bus IED (BIED), and 1 Under Frequency Load Shedding IED (UFIED) as shown in Figure 6.3. In this redundantly designed system, the voltage transformers located at each 66kV bus reduce the voltage to an 11kV level, which is typically used for distribution purposes. The substation is interconnected with neighboring substations through line feeders, ensuring a resilient configuration. To simulate GOOSE communications, a unique MAC address is assigned to each IED, and it is presumed that all 18 IEDs belong to the same multicast group and can receive multicast frames sent by any IED. Then, the power system data log is created manually for each IED in CSV format to describe the operating current, voltage, power, and frequency measurements under various scenarios.

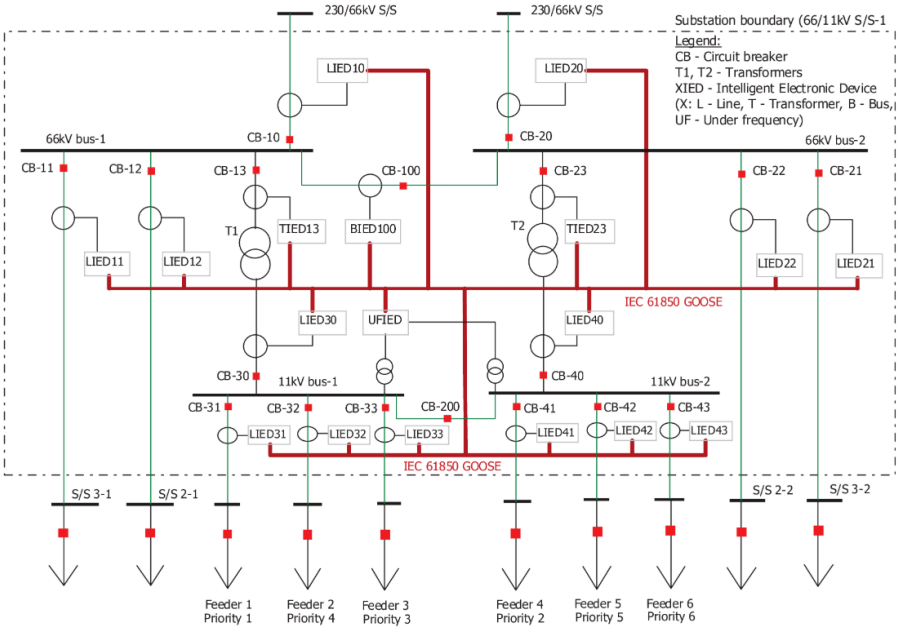


Figure 6.3: Single-line diagram of the 66/11kV substation automation system

Figure 6.4 illustrates the communication workflow in this test system. Firstly, the SCL conversion program extracts meaningful IED models from the SCL file to generate attack-free GOOSE traces. The Attack-Free Trace Generator accepts power system data logs, simulation configurations, and static IED models. The PowerWorld simulator generates the power system data log with IED nominal current and voltage measurements in time series order. The scenario configuration, power system configuration, and simulation configuration define the scenario setup, power system setup and the simulation setup. The Attack-Free Trace Generator generates GOOSE traces from these inputs. To generate attack-induced GOOSE traces the Attack-Induced Trace Generator requires a network trace and attack scenario configuration. The Attack-Free Trace Generator generates the input network trace. Then the Attack-Induced Trace Generator's traffic replay tool reproduces the input network trace's traffic as a baseline for editing. The Attack-Induced Trace generator injects specific attack signatures into the input network trace by defining the attack type, the IED GOOSE identifier, the time of attack, and the value to be modified in the attack scenario settings.

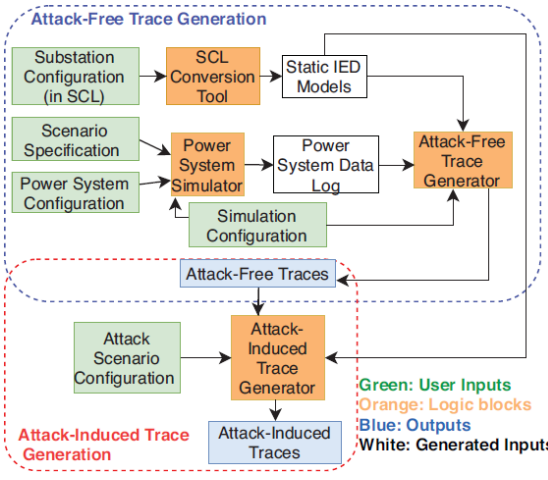


Figure 6.4: Communication framework for generating attack-free and attack induced GOOSE traces.

6.3. Scenario description of the Test Systems dataset

The test systems generated two datasets – 4 IED dataset from test system 1 and 18 IED dataset from test system 2. These datasets are used to evaluate the effectiveness of the proposed approach. Generally, there are three different behaviors in substation automation, which are: normal operation when no unusual events happen, disturbance operation when non-malicious events happen and attack operations that disrupts the substations operation and cause damage.

6.3.1. Test System 1: 4-IED dataset

In this dataset, the network 'pcap' files contains the network features highlighted in Section 5.3.1. The sensor data only contains circuit current values from the four IEDs and operational status of the various circuit breaker. The 'pcap' files from the five VMs were converted to comma-separated values (CSV) files and merged into one CSV file. This CSV file is then linked with the physical sensor data to create the dataset, which contains both network features and physical features. In this thesis, the 4-IED dataset samples consist of four types of behaviour. The following are the scenarios in the dataset:

1. Normal operations: No unusual event occurs.
2. Disturbance operation: A fault in the phase-to-phase connection, which is related to the failure of the overcurrent protection, and this failure leads to the breaker failure protection being activated.
3. Cyberattacks from IED 1 under normal and disturbance operation: IED1 is used to describe the IED protecting transformer 1 i.e., transformer 1(IED_PIOC_TRSF1). Two attack scenarios regarding GOOSE messages are created from IED1.

- Replay attack: IED 1 injects replayed GOOSE trip messages and stop protection or trigger unexpected protection. Figure 6.5 shows the Wireshark capture of replay attacks from IED 1 (IED_POIC_TRSF1).

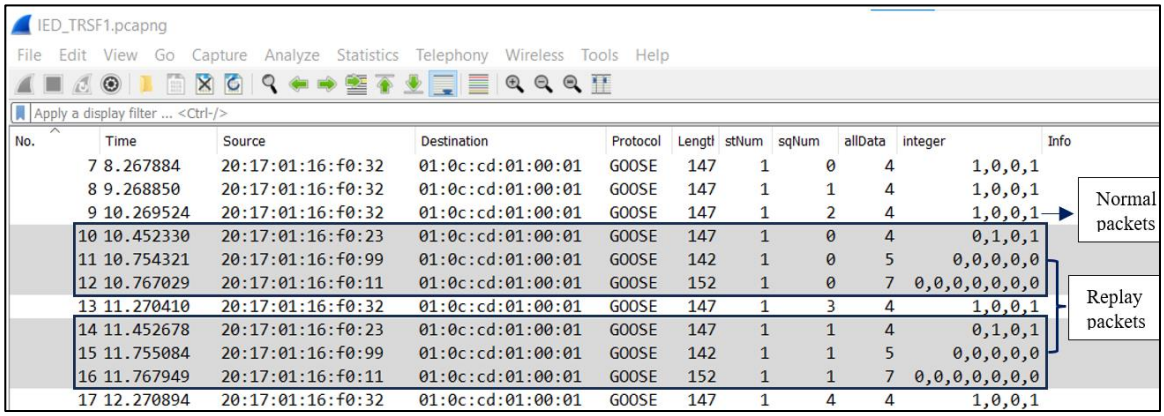


Figure 6.5: Wireshark capture of replay attack GOOSE packets from IED 1.

- FDIA: Original non-trip messages from IED1 are modified to trip messages and to stop protection or trigger unexpected protection. Figure 6.6 shows the Wireshark capture of FDIA attacks from IED 1 (IED_POIC_TRSF1).

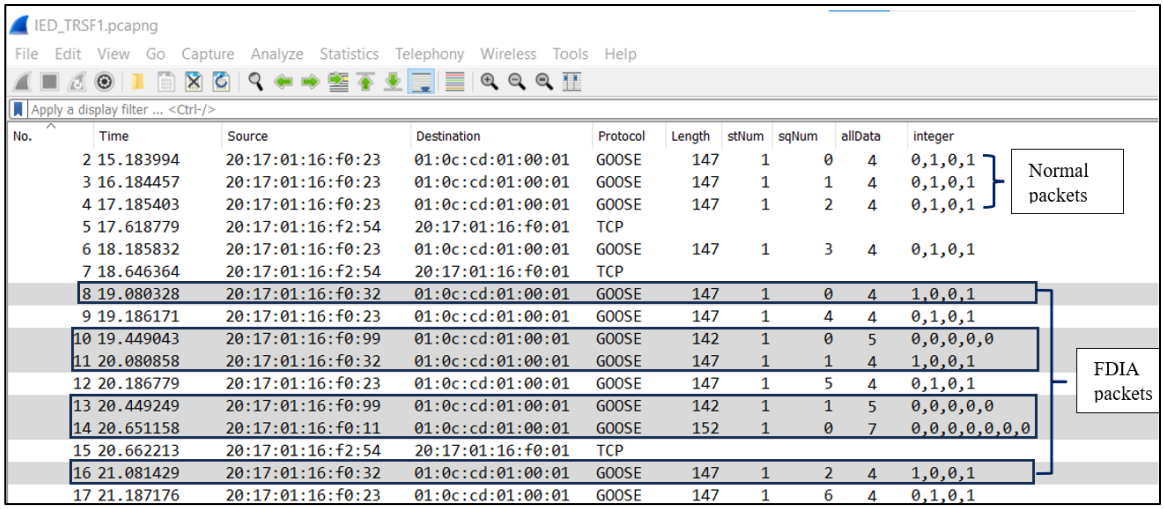


Figure 6.6: Wireshark capture of FDIA GOOSE packets from IED 1.

4. Cyberattacks from IED 2 under normal and disturbance operation: IED2 is used to describe the IED protecting transformer 2 i.e. transformer 2 (IED_PIOC_TRSF2). Two attack scenarios regarding GOOSE messages are created from IED2.

    - Replay attack: IED 2 injects replayed GOOSE trip messages and stop protection or trigger unexpected protection. Figure 6.7 shows the Wireshark capture of replay attacks from IED 1 (IED_POIC_TRSF2).

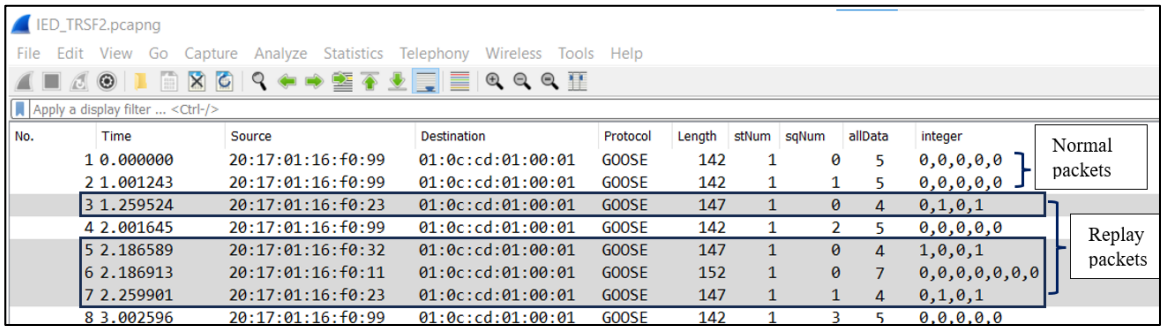


Figure 6.7: Wireshark capture of replay attack GOOSE packets from IED 2.

- FDIA: Original non-trip messages from IED2 are modified to trip messages and to stop protection or trigger unexpected protection. Figure 6.8 shows the Wireshark capture of FDIA attacks from IED 1 (IED_POIC_TRSF2).

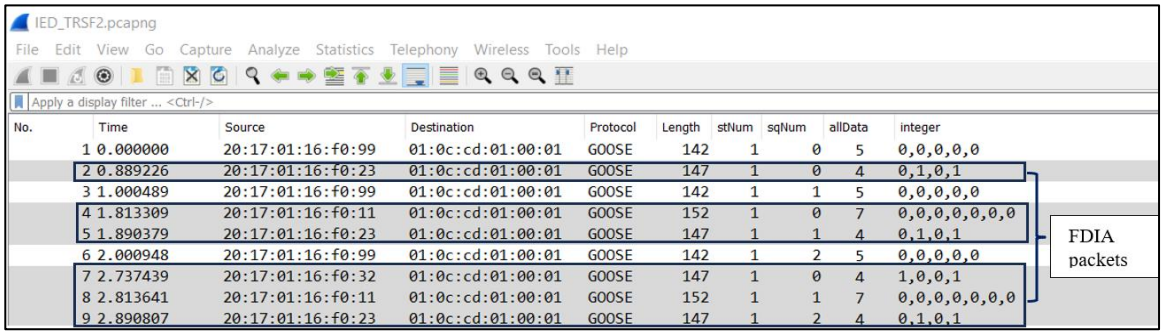


Figure 6.8: Wireshark capture of FDIA GOOSE packets from IED 2.

Based on the case description of the 4-IED dataset, only two IEDs - IED1 and IED2, account for the attack instances. Table 6.1 lists the sample distribution of the different scenarios created from test system 1 to generate the 4 IED dataset.

Table 6.1: Scenario description of the 4-IED dataset

Behavior type	Number of Samples	Labels
Normal Operation	7447	Normal
Disturbance Operation	12457	Disturbance
Attacks from IED 1 under normal and disturbance operation	4232 3783	Replay FDIA
Attack from IED 2 under normal and disturbance operation	5078 4824	Replay FDIA

6.3.2. Test System 2: 18-IED dataset

In this dataset, the power system data log for each IED is generated in CSV format to describe operating current, voltage, power, and frequency measurements during normal and disturbance operations. According to Figure 6.3 each of the 18 IEDs is assigned a unique MAC address to simulate GOOSE communications. After simulation, the 'pcap' files from each IED are converted to comma-separated values (CSV) files. The CSV format of the GOOSE communication of each IED is then merged with the power system data log of the respective IED. In this thesis, the 4-IED dataset samples consist of six scenarios. The following are the scenarios in the dataset:

1. Normal operations: This operation experienced two scenarios – variable and non-variable load circumstances. During the variable load scenario there is a demand

shift over a period of time causing each IED to display distinct current and power measurements. Conversely, during the non-variable load scenario there is a stable energy flow as a result of negligible variations in load demands.

2. Disturbance operations: Three disturbance scenarios in which the substation protection system operates are considered.
    - Busbar protection: This disturbance operation is the inability of IED/IEDs to detect overcurrent. In this scenario, a fault arises at the 66kV bus-1 busbar, and LIED10 detects an overcurrent while other IEDs fail to detect it. LIED10, through GOOSE communication then identifies the busbar fault, initiates a trip for its breaker and associated busbar breakers. The trip status is then transmitted to LIED11, LIED12 AND TIED13.
    - Breaker failure: This disturbance operation constitutes a circuit breaker experiencing mechanical failure. In this scenario, a fault arises in the feeder connecting substation S/S 3-1, activating the associated LIED11 overcurrent element. However, a mechanical failure prevents breaker CB-11 from tripping. Alternatively, the GOOSE communication from LIED11 to LIED10, LIED12, and TIED13 results in the tripping of circuit breakers CB-10, CB-12, and CB-13, as well as the remote circuit breaker in S/S 3-1.
    - Under frequency: In this disturbance scenario, there is a frequency drop across the busbar. The Under Frequency Intelligent Electronic Device (UFIED) detects frequency drops in the 11kV buses via GOOSE. It then initiates a trip sequence starting with the least priority consumer and

progressing through higher priority consumers until the frequency stabilizes.

The Following is the description of the simulated GOOSE-related cyberattack operations designed to compromise the substation's operation.

3. Data Manipulation: This involves the spoofing of false information to the IEDs to mask unauthorized changes. Below are the tactics in which this attack was conducted.
    - Data Manipulation1 (DM1): In this attack scenario false current measurements are injected to bias the power system state estimation process without being detected as shown in Figure 6.9, Figure 6.10, and Figure 6.11.

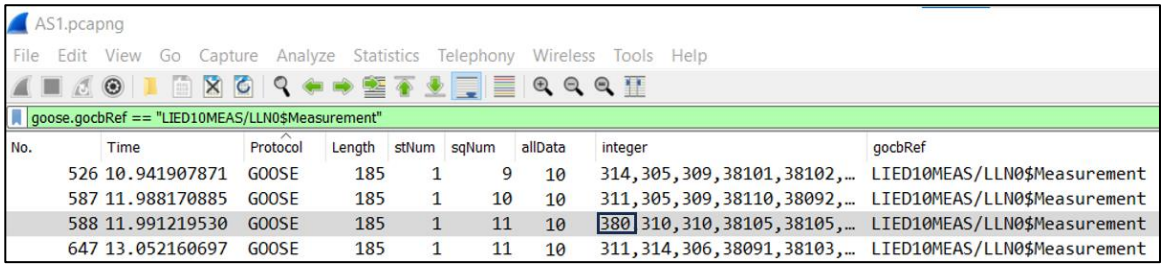


Figure 6.9: LIED10 injects a GOOSE frame (No. 588) with a value of 380 for phase A current magnitude at 11.9 seconds.

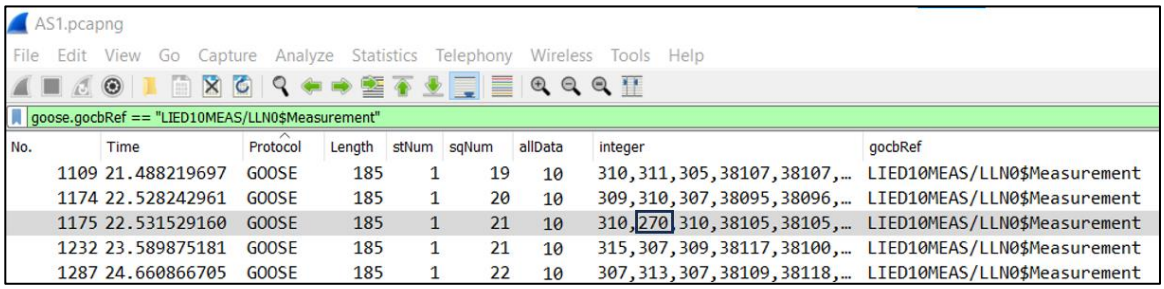


Figure 6.10: LIED10 injects a GOOSE frame (No. 1175) with a value of 270 for phase B current magnitude at 22.5 seconds.

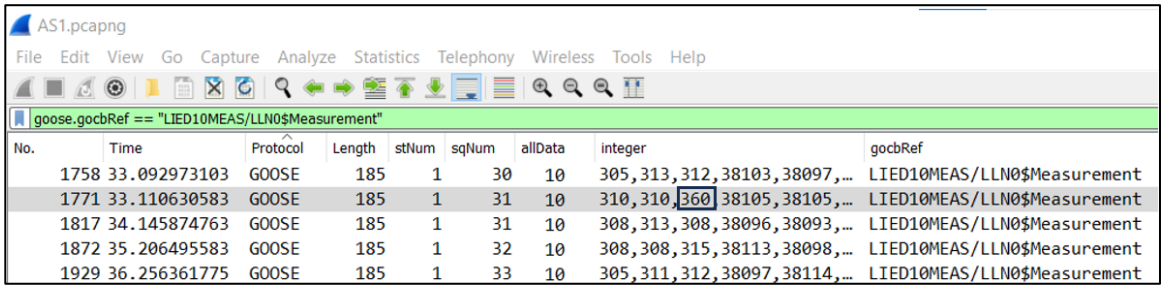


Figure 6.11: LIED10 injects a GOOSE frame (No. 1771) with a value of 360 for phase C current magnitude at 33.1 seconds.

- Data Manipulation2 (DM2): In this attack scenario, a malicious GOOSE frame is injected to control the state of the circuit breaker as shown in Figure 6.12.

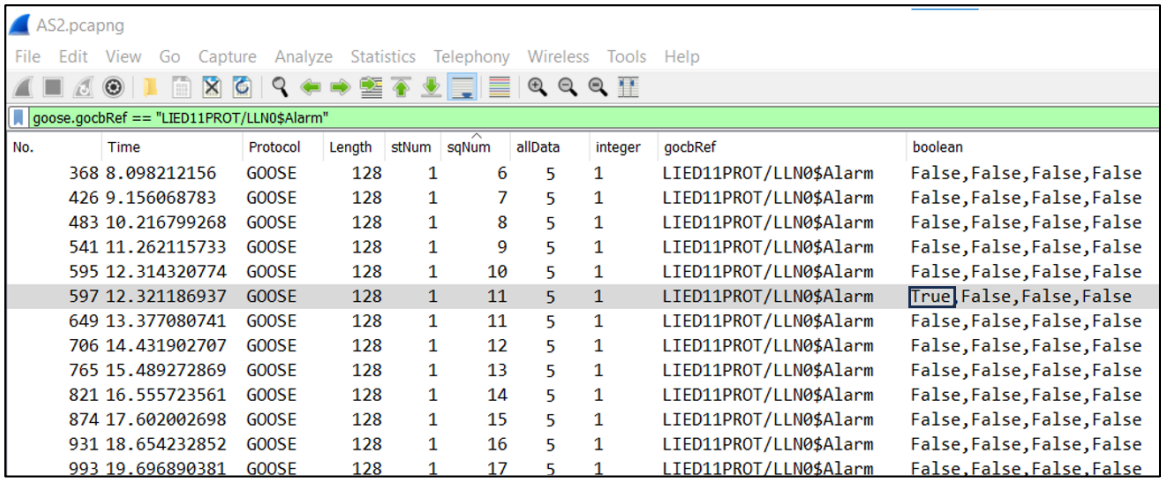
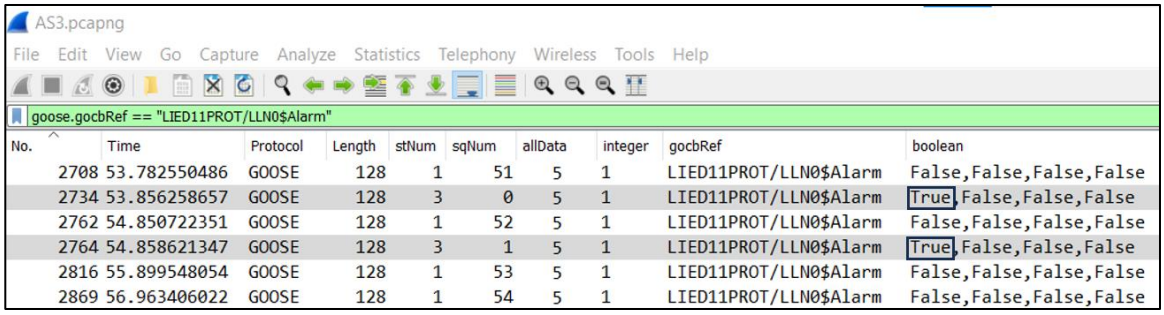


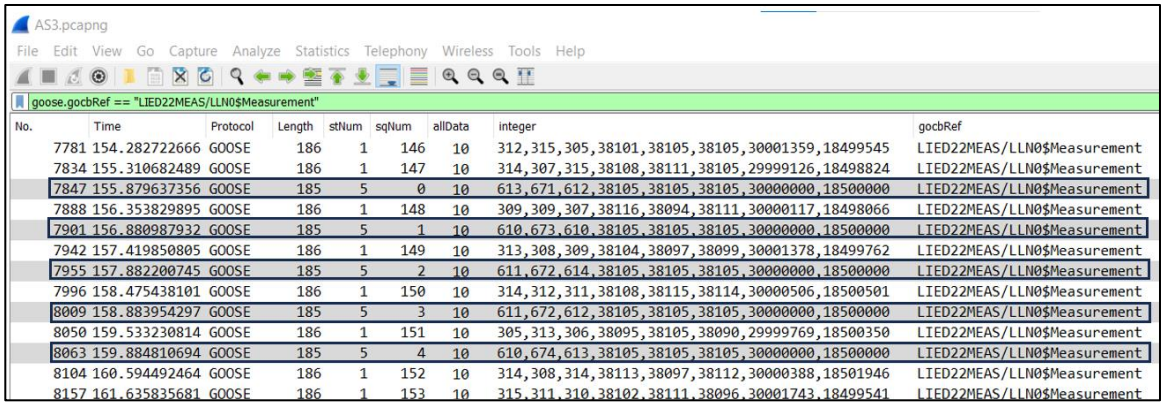
Figure 6.12: LIED11 injects malicious GOOSE frame (No. 597) changing the circuit breaker status from FALSE to TRUE ('tripped') at 12.3 seconds.

- Data Manipulation3 (DM3): In this attack scenario, an old GOOSE payload containing circuit breaker 'trip' status and other measurements messages is replayed as shown in Figure 6.13 and Figure 6.14.



No.	Time	Protocol	Length	stNum	sqNum	allData	integer	gocbRef	boolean
2708	53.782550486	GOOSE	128	1	51	5	1	LIED11PROT/LLN0$Alarm	False,False,False,False
2734	53.856258657	GOOSE	128	3	0	5	1	LIED11PROT/LLN0$Alarm	True,False,False,False
2762	54.850722351	GOOSE	128	1	52	5	1	LIED11PROT/LLN0$Alarm	False,False,False,False
2764	54.858621347	GOOSE	128	3	1	5	1	LIED11PROT/LLN0$Alarm	True,False,False,False
2816	55.899548054	GOOSE	128	1	53	5	1	LIED11PROT/LLN0$Alarm	False,False,False,False
2869	56.963406022	GOOSE	128	1	54	5	1	LIED11PROT/LLN0$Alarm	False,False,False,False

Figure 6.13: LIED11 replays valid GOOSE frames (No. 2734 and No. 2764) with "open" (True) circuit breaker data at times 53.8 sec and 54.8 sec.



No.	Time	Protocol	Length	stNum	sqNum	allData	integer	gocbRef
7781	154.282722666	GOOSE	186	1	146	10	312,315,305,38101,38105,38105,30001359,18499545	LIED22MEAS/LLN0$Measurement
7834	155.310682489	GOOSE	186	1	147	10	314,307,315,38108,38111,38105,29999126,18498824	LIED22MEAS/LLN0$Measurement
7847	155.879637356	GOOSE	185	5	0	10	613,671,612,38105,38105,38105,30000000,18500000	LIED22MEAS/LLN0$Measurement
7888	156.353829895	GOOSE	186	1	148	10	309,309,307,38116,38094,38111,30000117,18498066	LIED22MEAS/LLN0$Measurement
7901	156.880987932	GOOSE	185	5	1	10	610,673,610,38105,38105,38105,30000000,18500000	LIED22MEAS/LLN0$Measurement
7942	157.419850805	GOOSE	186	1	149	10	313,308,309,38104,38097,38099,30001378,18499762	LIED22MEAS/LLN0$Measurement
7955	157.882200745	GOOSE	185	5	2	10	611,672,614,38105,38105,38105,30000000,18500000	LIED22MEAS/LLN0$Measurement
7996	158.475438101	GOOSE	186	1	150	10	314,312,311,38108,38115,38114,30000506,18500501	LIED22MEAS/LLN0$Measurement
8009	158.883954297	GOOSE	185	5	3	10	611,672,612,38105,38105,38105,30000000,18500000	LIED22MEAS/LLN0$Measurement
8050	159.533230814	GOOSE	186	1	151	10	305,313,306,38095,38105,38090,29999769,18500350	LIED22MEAS/LLN0$Measurement
8063	159.884810694	GOOSE	185	5	4	10	610,674,613,38105,38105,38105,30000000,18500000	LIED22MEAS/LLN0$Measurement
8104	160.594492464	GOOSE	186	1	152	10	314,308,314,38113,38097,38112,30000388,18501946	LIED22MEAS/LLN0$Measurement
8157	161.635835681	GOOSE	186	1	153	10	315,311,310,38102,38111,38096,30001743,18499541	LIED22MEAS/LLN0$Measurement

Figure 6.14: LIED22 replays fault current measurements (No. 7847, No. 7901, No. 7955, No. 8009, and No. 8063) between times 155.8 sec and 159.88 sec.

4. Denial of Service (DoS): The objective of this attack scenario is to obstruct the flow of information to the intended IEDs by overwhelming the substation network with GOOSE messages to reduce service availability. Figure 6.15 shows the DoS attack on LIED10.

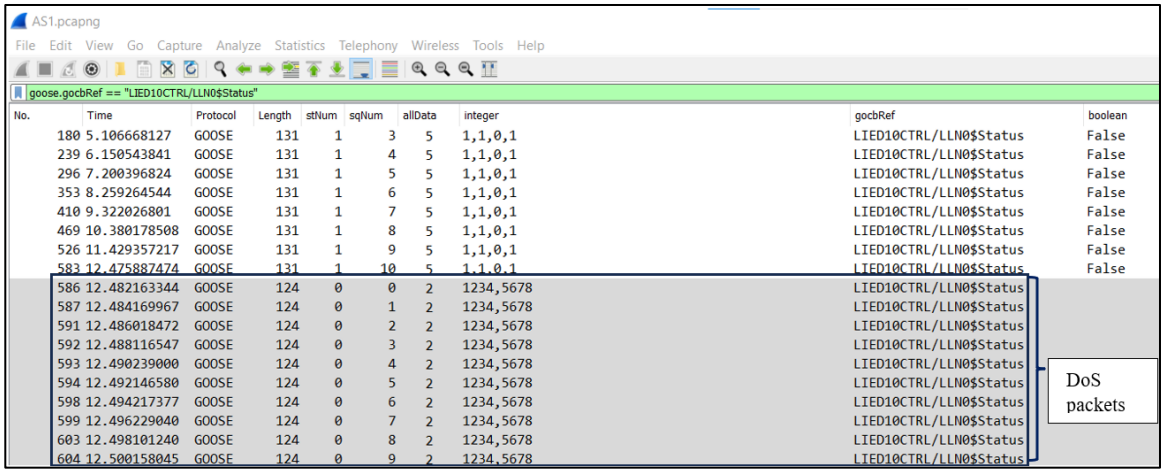


Figure 6.15: Denial-of-Service (DoS) attack on LIED10

5. Message Suppression (MS): This attack involves modifying the GOOSE header fields to take over the communication channel in order to prevent legitimate IEDs from receiving vital messages or updates. Below are the tactics in which this attack was conducted.

    - Message Suppression1 (MS1): In this attack scenario, a high Stnum value or slightly higher than the previously recorded Stnum is injected, where Sqnum ≠ 0 as shown in Figure 6.16 and Figure 6.17.

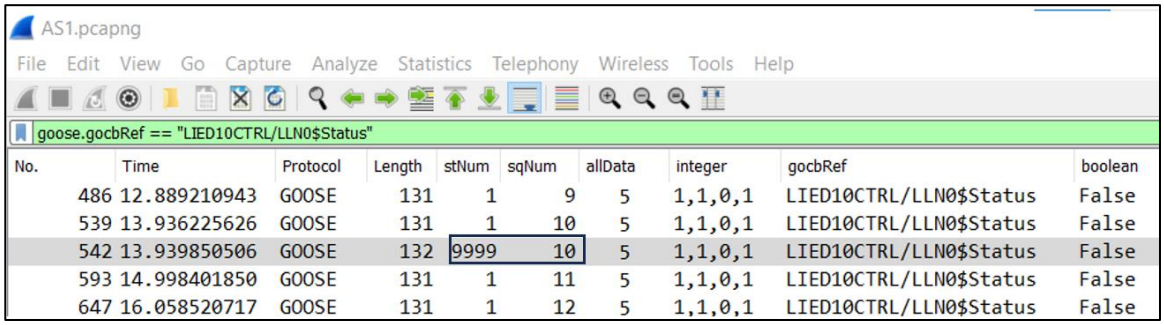
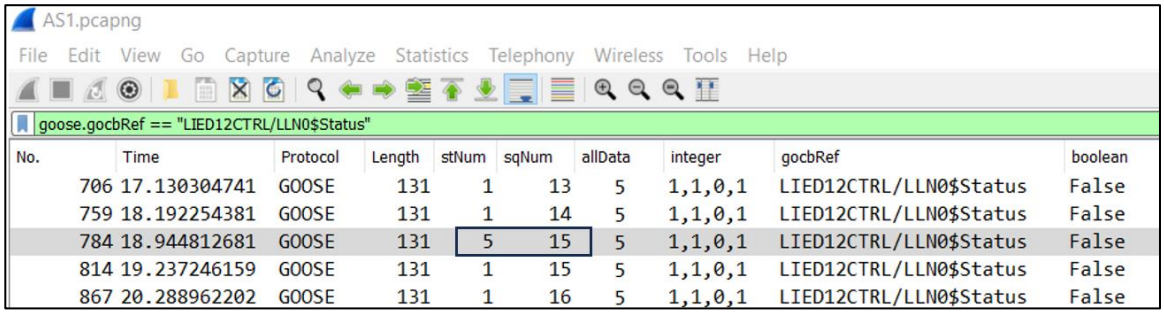


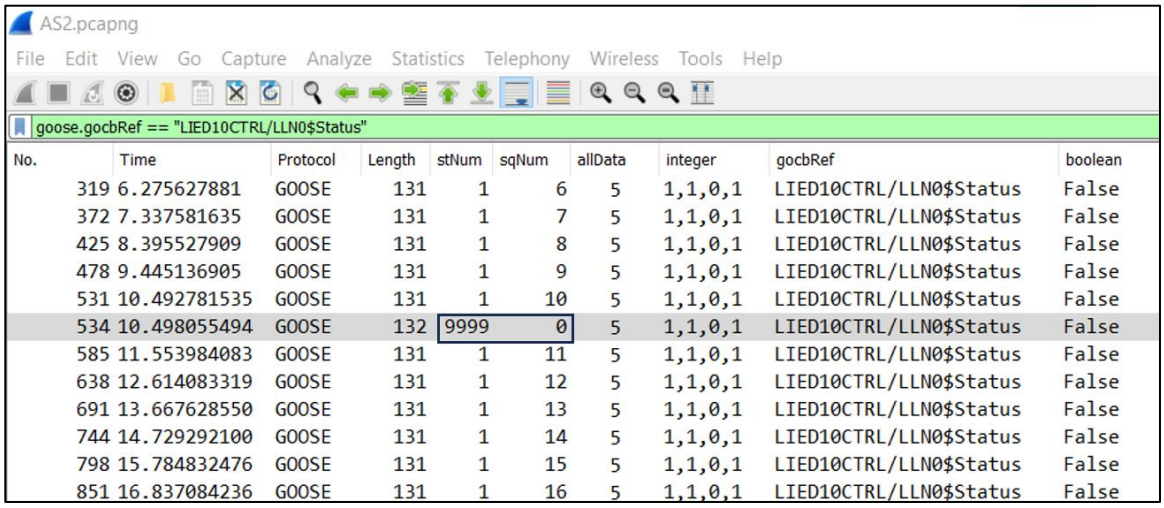
Figure 6.16: LIED10 injects a GOOSE frame (No. 542) with Stnum=9999 and Sqnum=10 at 13.9 secs.



No.	Time	Protocol	Length	stNum	sqNum	allData	integer	gocbRef	boolean
706	17.130304741	GOOSE	131	1	13	5	1,1,0,1	LIED12CTRL/LLN0$Status	False
759	18.192254381	GOOSE	131	1	14	5	1,1,0,1	LIED12CTRL/LLN0$Status	False
784	18.944812681	GOOSE	131	5	15	5	1,1,0,1	LIED12CTRL/LLN0$Status	False
814	19.237246159	GOOSE	131	1	15	5	1,1,0,1	LIED12CTRL/LLN0$Status	False
867	20.288962202	GOOSE	131	1	16	5	1,1,0,1	LIED12CTRL/LLN0$Status	False

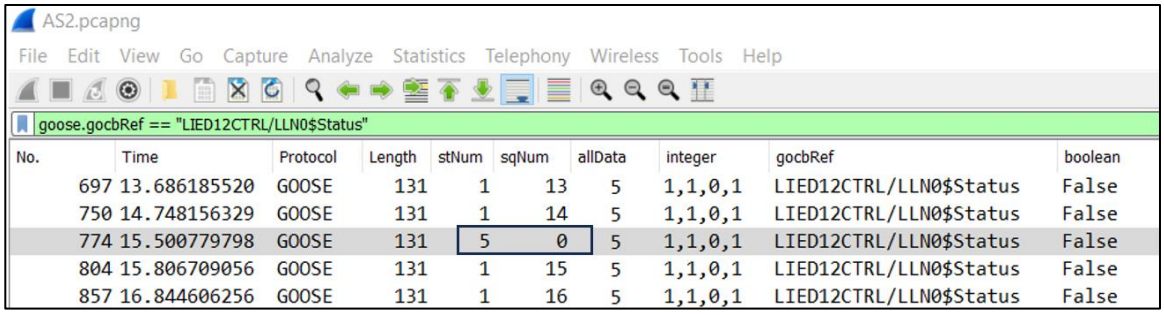
Figure 6.17: LIED10 injects a GOOSE frame (No. 784) with Stnum=5 and Sqnum=15 at 18.9 secs.

- Message Suppression2 (MS2): In this attack scenario, a previously valid GOOSE frame containing high Stnum is replayed, where Sqnum = 0 but stale timestamp as shown in Figure 6.18 and Figure 6.19.



No.	Time	Protocol	Length	stNum	sqNum	allData	integer	gocbRef	boolean
319	6.275627881	GOOSE	131	1	6	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
372	7.337581635	GOOSE	131	1	7	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
425	8.395527909	GOOSE	131	1	8	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
478	9.445136905	GOOSE	131	1	9	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
531	10.492781535	GOOSE	131	1	10	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
534	10.498055494	GOOSE	132	9999	0	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
585	11.553984083	GOOSE	131	1	11	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
638	12.614083319	GOOSE	131	1	12	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
691	13.667628550	GOOSE	131	1	13	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
744	14.729292100	GOOSE	131	1	14	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
798	15.784832476	GOOSE	131	1	15	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
851	16.837084236	GOOSE	131	1	16	5	1,1,0,1	LIED10CTRL/LLN0$Status	False

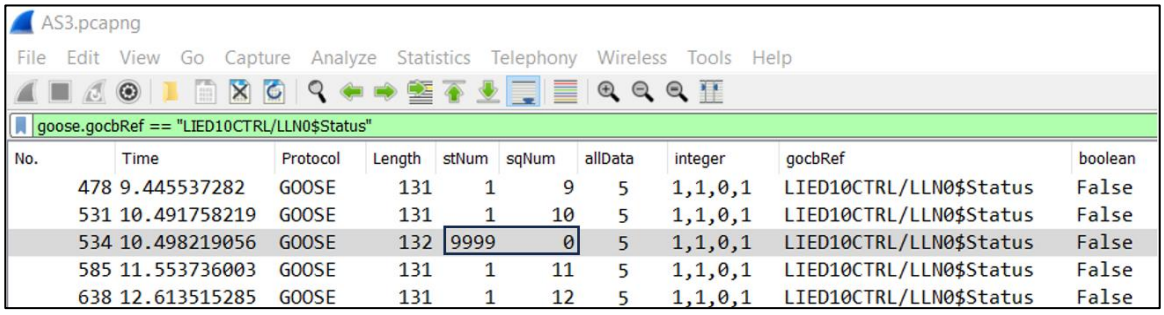
Figure 6.18: LIED10 replays a GOOSE frame (No. 534) with Stnum=9999 and Sqnum=0 at 10.4 secs.



No.	Time	Protocol	Length	stNum	sqNum	allData	integer	gocbRef	boolean
697	13.686185520	GOOSE	131	1	13	5	1,1,0,1	LIED12CTRL/LLN0$Status	False
750	14.748156329	GOOSE	131	1	14	5	1,1,0,1	LIED12CTRL/LLN0$Status	False
774	15.500779798	GOOSE	131	5	0	5	1,1,0,1	LIED12CTRL/LLN0$Status	False
804	15.806709056	GOOSE	131	1	15	5	1,1,0,1	LIED12CTRL/LLN0$Status	False
857	16.844606256	GOOSE	131	1	16	5	1,1,0,1	LIED12CTRL/LLN0$Status	False

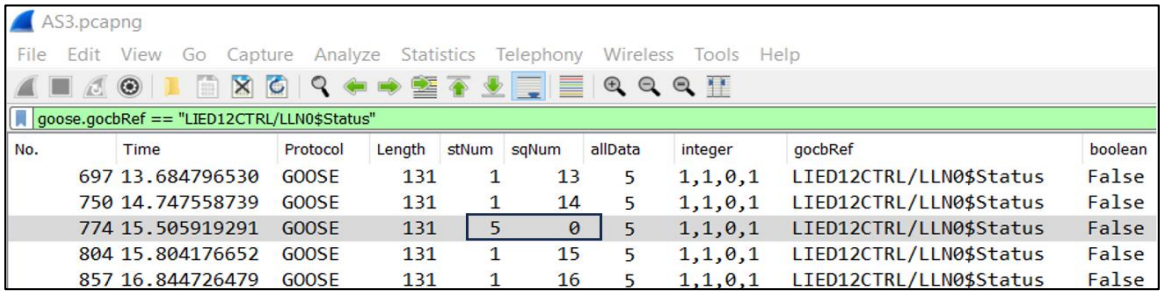
Figure 6.19: LIED12 replays a GOOSE frame (No. 774) with Stnum=5 and Sqnum=0 at 15.5 secs.

- Message Suppression3 (MS3): In this attack scenario, a high Stnum frame with Sqnum = 0 and a valid timestamp is injected, as shown in Figure 6.20 and Figure 6.21.



No.	Time	Protocol	Length	stNum	sqNum	allData	integer	gocbRef	boolean
478	9.445537282	GOOSE	131	1	9	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
531	10.491758219	GOOSE	131	1	10	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
534	10.498219056	GOOSE	132	9999	0	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
585	11.553736003	GOOSE	131	1	11	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
638	12.613515285	GOOSE	131	1	12	5	1,1,0,1	LIED10CTRL/LLN0$Status	False

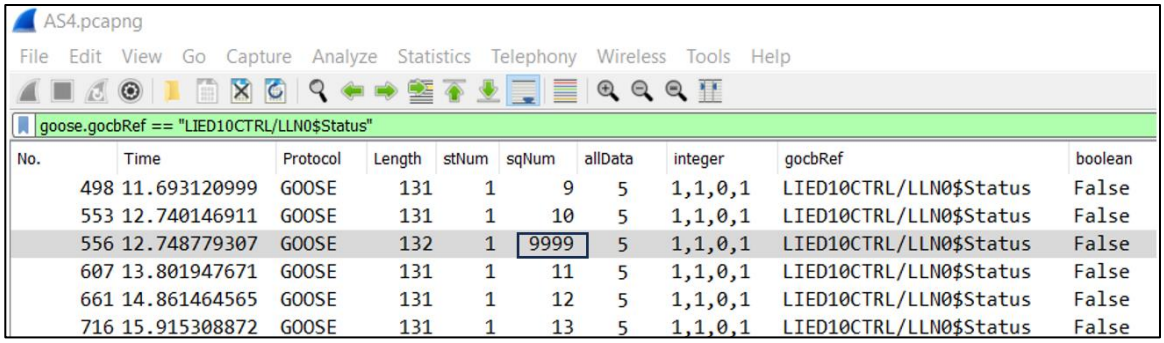
Figure 6.20: LIED10 injects a GOOSE frame (No. 534) with Stnum=9999 and Sqnum=0 at 10.4 secs.



No.	Time	Protocol	Length	stNum	sqNum	allData	integer	gocbRef	boolean
697	13.684796530	GOOSE	131	1	13	5	1,1,0,1	LIED12CTRL/LLN0$Status	False
750	14.747558739	GOOSE	131	1	14	5	1,1,0,1	LIED12CTRL/LLN0$Status	False
774	15.505919291	GOOSE	131	5	0	5	1,1,0,1	LIED12CTRL/LLN0$Status	False
804	15.804176652	GOOSE	131	1	15	5	1,1,0,1	LIED12CTRL/LLN0$Status	False
857	16.844726479	GOOSE	131	1	16	5	1,1,0,1	LIED12CTRL/LLN0$Status	False

Figure 6.21: LIED12 injects a GOOSE frame (No. 774) with Stnum=5 and Sqnum=0 at 15.5 secs.

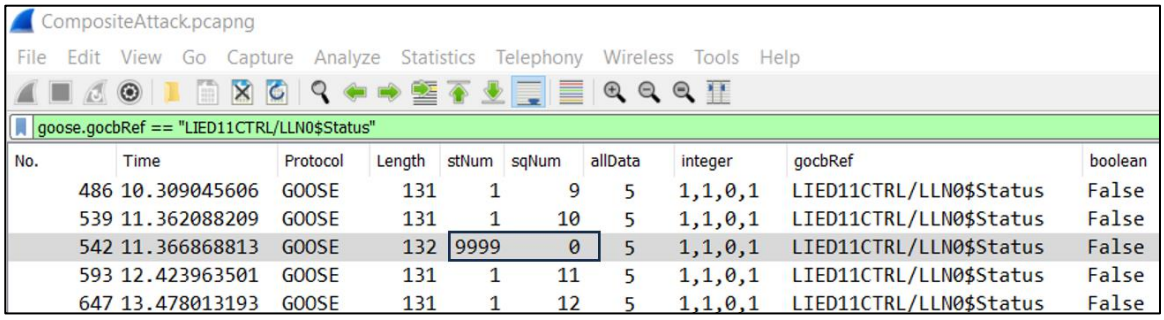
- Message Suppression4 (MS4): In this attack scenario, a high Sqnum frame to cause GOOSE frames to arrive at the receiver out-of-sequence i.e., not matching the order of transmission at the sender is injected, as shown in Figure 6.22.



No.	Time	Protocol	Length	stNum	sqNum	allData	integer	gocbRef	boolean
498	11.693120999	GOOSE	131	1	9	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
553	12.740146911	GOOSE	131	1	10	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
556	12.748779307	GOOSE	132	1	9999	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
607	13.801947671	GOOSE	131	1	11	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
661	14.861464565	GOOSE	131	1	12	5	1,1,0,1	LIED10CTRL/LLN0$Status	False
716	15.915308872	GOOSE	131	1	13	5	1,1,0,1	LIED10CTRL/LLN0$Status	False

Figure 6.22: LIED10 injects a GOOSE frame (No. 556) with Sqnum=9999 at time= 12.7 sec.

6. Composite attack: This attack scenario is comprised of both a data manipulation attack and a message suppression attack as shown in Figure 6.23 and Figure 6.24. A high Stnum attack is injected followed by the modification of the circuit breaker status associated with CB-11.



No.	Time	Protocol	Length	stNum	sqNum	allData	integer	gocbRef	boolean
486	10.309045606	GOOSE	131	1	9	5	1,1,0,1	LIED11CTRL/LLN0$Status	False
539	11.362088209	GOOSE	131	1	10	5	1,1,0,1	LIED11CTRL/LLN0$Status	False
542	11.366868813	GOOSE	132	9999	0	5	1,1,0,1	LIED11CTRL/LLN0$Status	False
593	12.423963501	GOOSE	131	1	11	5	1,1,0,1	LIED11CTRL/LLN0$Status	False
647	13.478013193	GOOSE	131	1	12	5	1,1,0,1	LIED11CTRL/LLN0$Status	False

Figure 6.23: LIED11 injects a GOOSE frame (No. 542) with Stnum=9999 and Sqnum=0 at 11.3 sec.

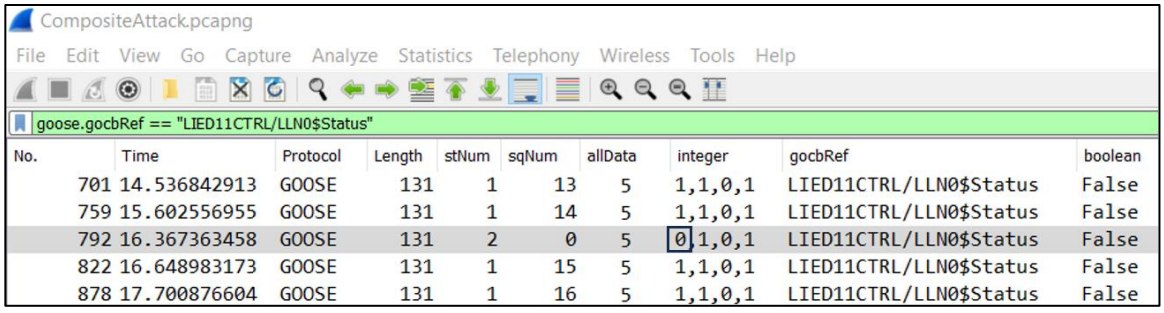


Figure 6.24: LIED11 modifies the CB-11 Boolean value from '1' to '0' and injects the modified GOOSE frame (No. 792) at 16.3 secs.

From the case description of the 18-IED dataset only four IEDs bearing the designations LIED 10, LIED 11, LIED 12, and LIED 22 out of the 18 IEDs account for most attack instances. Table 6.2 shows the total number of attacks identified in these IED datasets by aggregating the number of attack packets encountered during each IED attack. It also presents a comprehensive record of the overall count of attacks that were detected in the IED dataset by summing up the number of attack packets encountered during each individual attack on an IED.

Table 6.2: Scenario description of the 18 IED dataset

IED	Packets per attack	Total packets
LIED 10	• DM = 3 packets • DoS = 5000 packets • MS = 4 packets	5007
LIED 11	• DM = 3 packets • Composite = 2 packets (1 DM and 1 MS)	5
LIED 12	• DoS = 5000 packets • MS = 3	5003
LIED 22	• DM = 5 packets	5

6.4.Training and Testing of Automated Classification

Various combinations of IED datasets are tuned to evaluate the sensitivity of the proposed approach to selecting datasets for training and testing. When these combinations are configured, the dataset used for training contains normal, disturbance and all the cyberattack cases while the dataset used for testing consists of normal scenarios, disturbance scenarios as well as any of the cyberattack cases. This assembly method ensures that the full spectrum of all conditions is covered, allowing a more comprehensive assessment of the response of the proposed approach.

These combinations are applied to the two test system datasets. In the 4-IED dataset, two cases are used in this thesis. The first case is aimed at training the models for all normal, disturbance, FDIA attacks and replay attacks originating exclusively from IED1 in order to detect and classify FDIA attacks and replay attacks originating from IED2. In the second case, the data sets for training and testing are then swapped. For the 18-IED dataset, four cases representing different combinations of the line IEDs (LIEDs) are used. Table 6.3 highlights the combinations for training and testing.

Table 6.3: 18-IED Dataset Combinations for Training and Testing

Case Number	Line IED Combinations	
	Training	Testing
Case 1	LIED 10 & LIED 11	LIED 12 & LIED 22
Case 2	LIED 12 & LIED 22	LIED 10 & LIED 11
Case 3	LIED 10 & LIED 22	LIED 11 & LIED 12
Case 4	LIED 11 & LIED 12	LIED 10 & LIED 22

6.5. Results of Implementing the Proposed Approach on the 4-IED Dataset

The proposed Fine-Tree-Bagging Ensemble (FTBE) learning approach is implemented on the simulated substation automation system data of test system 1. The number of learners and k-fold values were determined via a trial-and-error method to obtain the highest accuracy, which was kept constant for further analysis. Furthermore, the results have been compared with those obtained when applying other classifiers highlighted in Chapter 3. Table 6.4 presents the detection accuracy obtained by implementing the proposed FTBE approach on the 4-IED dataset. The test accuracy of the decision tree (DT) approach decreased from 93.63% to 91.80% when the training and testing datasets were switched. Similarly, the accuracy of K-nearest neighbour (KNN) and support vector machine (SVM) classifiers decreased when the datasets were changed as well. The FTBE approach follows the same trend when the datasets are interchange, but it performs better than the other classifiers, exhibiting a maximum detection accuracy of 94.24% and 92.60%,

in case 1 and case 2 respectively. In case 1, the highest classification accuracy is achieved when k-fold is set to 6 and there are 10 learners. Conversely, case 2 achieved maximum classification accuracy with 4 learners and k-fold set to 12. The accuracy difference between case 1 and case 2 indicates a minor sensitivity to the selection of the training and the testing dataset. It can be deduced that the specific characteristics of the IEDs can affect the performance of the proposed approach. Observing the results, however, reveals that the high accuracy of the FTBE method remained comparatively stable even when the training and testing datasets were swapped. This consistency demonstrates the insensitivity of 4-IED datasets approach to selecting IEDs for training and testing.

Table 6.4: Comparative analysis of the Accuracy Results of the 4-IED Dataset

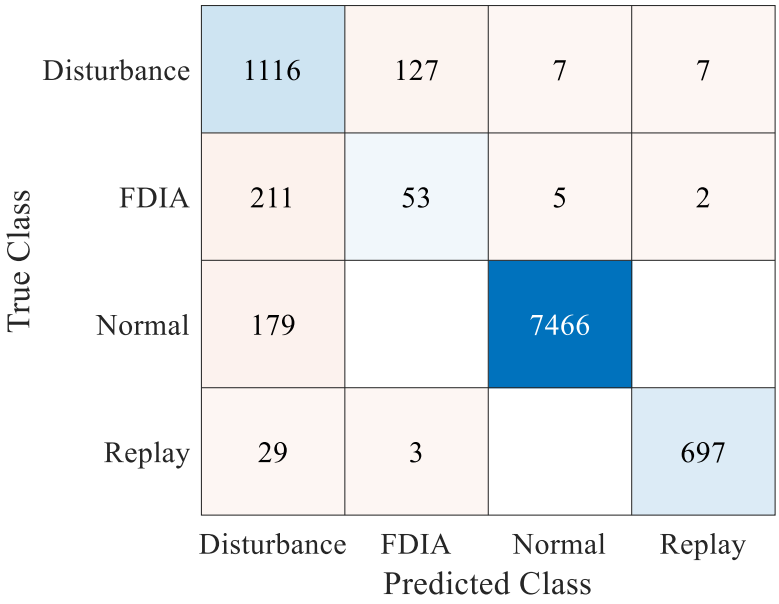
Case Number	Training	Testing	Methods	Train Accuracy (%)	Test Accuracy (%)
Case 1	IED 1	IED 2	DT	99.14	93.63
			KNN	98.12	92.65
			SVM	96.86	90.59
			FTBE	99.06	94.24
Case 2	IED 2	IED 1	DT	99.20	91.80
			KNN	99.40	91.43
			SVM	97.28	88.72
			FTBE	99.21	92.60

6.5.1. Confusion Matrices and Features Identification for Cyberattack Types

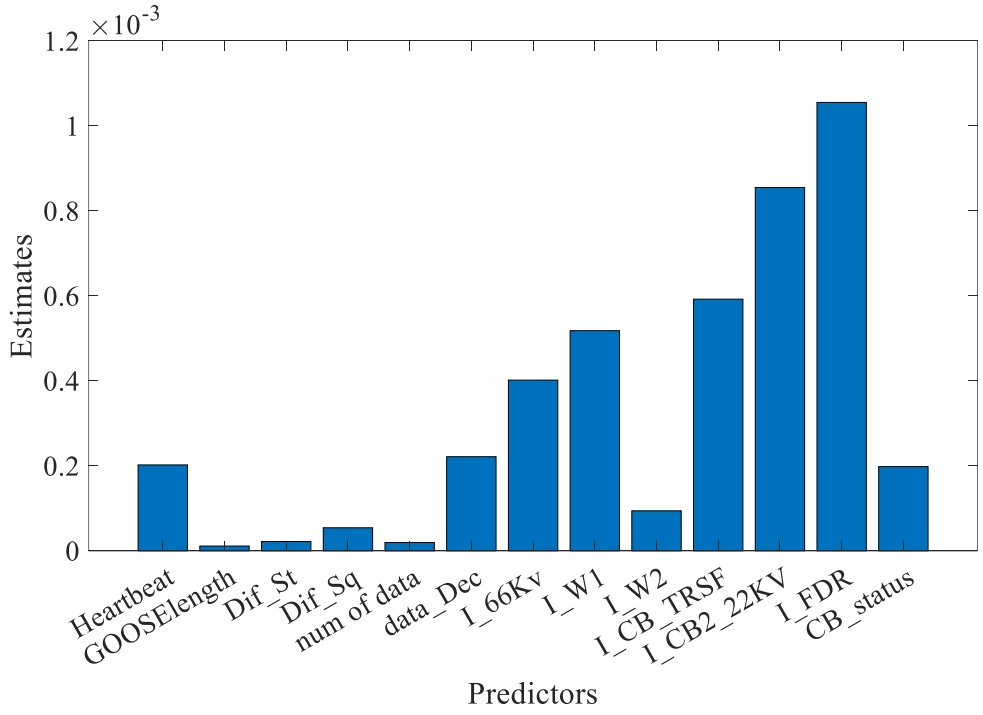
The detailed analysis of the confusion matrices and predictor importance for the 4 IED data set is provided in Figure 6.25 and Figure 6.26. The results obtained from the confusion matrices shown in Figure 6.25(a) and Figure 6.26(a) indicate that replay attacks are

relatively easier to identify compared to FDIA attacks. Specifically, in cases 1 and 2 the proposed approach is able to accurately detect replay attacks with an accuracy of 95.6% and 93.7% respectively. However, FDIA attacks were misclassified as normal or replay attacks for few instances and in some cases, they were misclassified as disturbances. Further examination of the predictor importance plots depicted in Figure 6.25(b) and Figure 6.26(b) reveals that among the physical features, the current at the feeders (I_FDR) had the highest PI estimate. In addition. Network features like heartbeat and Dec_allData showed similar PI estimate values, indicating their crucial role in classifying different types of cyberattacks.

Furthermore, it is observed that FDIA attacks are often followed by sudden changes in current or power consumption patterns. By continuously monitoring the feeder, the model can detect and classify such instances as false data being injected. The heartbeat and Dec_allData feature also play a significant role in identifying FDIA attacks. Any deviations in frequency, timing, or content of GOOSE packets suggest potential FDIA attacks. Furthermore, inconsistent or unexpected changes in Boolean control command signals (Dec-allData) also indicate an FDIA attack according to the learning approach. The "Dec allData" feature is highly important when it comes to classifying replay attacks. Any irregularities or patterns that do not adhere to typical characteristics imply that previously recorded control commands are being replayed, enabling the model to identify such attacks. However, changes in current values at the feeder also provide an indication of the impact of replay attacks. When previously recorded data is fed into the system with an attempt to appear as normal, deviations in current values from expected patterns can be an indicator that there is a replay attack.

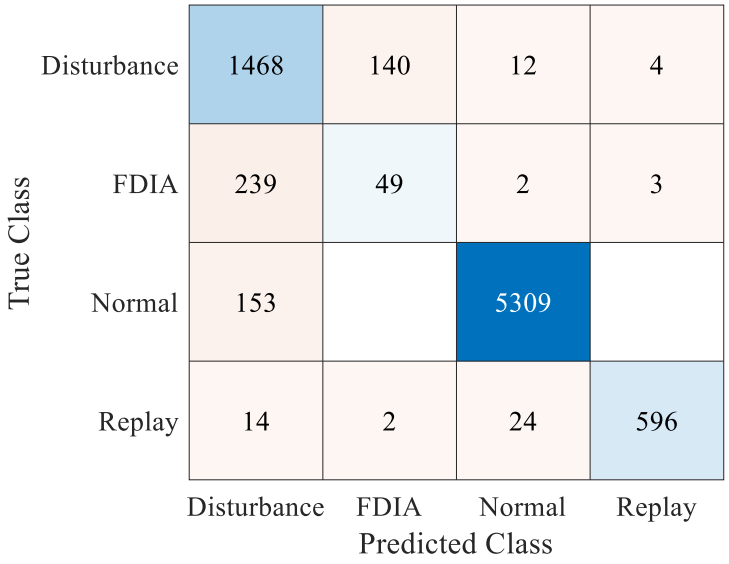


(a)

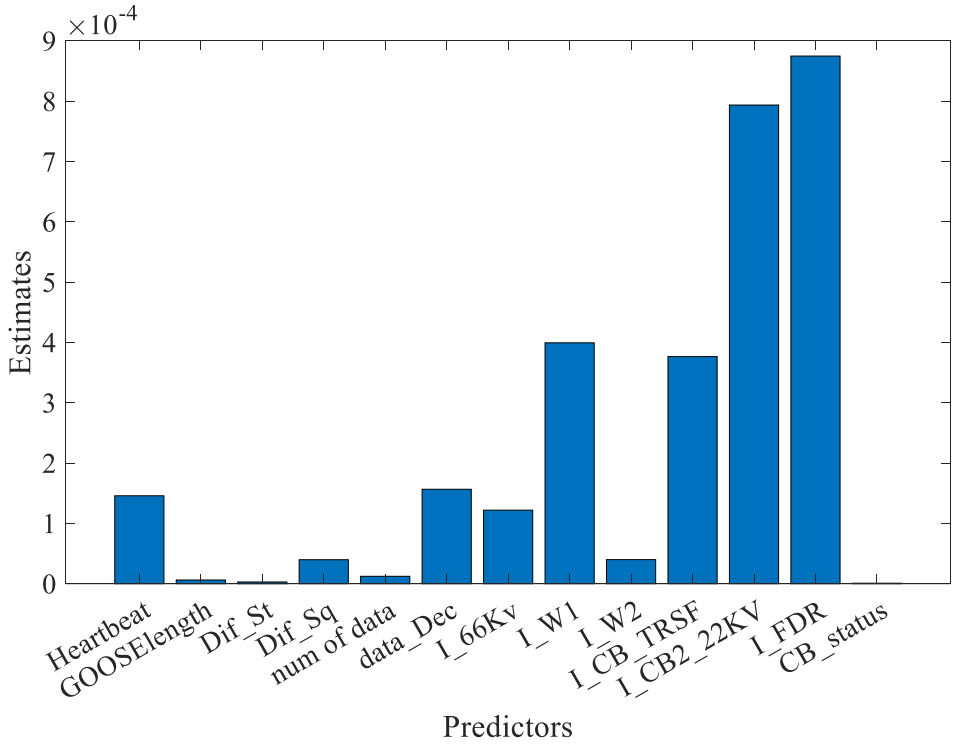


(b)

Figure 6.25: Classification results for case 1. (a) Confusion Matrix (b) predictor importance.



(a)



(b)

Figure 6.26: Classification results for case 2. (a) Confusion Matrix (b) predictor importance.

6.5.2. Sensitivity and Error Analysis

In this thesis a comprehensive study is conducted to examine how the proposed approach is influenced by the choice of the k-fold and number of learners parameters. The findings indicate that case 1 outperforms case 2 slightly in terms of accuracy. Using brute force, the highest accuracy for case 1 was achieved with 6-folds and 10 learners, while for case 2, it was with 12-folds and 4 learners. However, it is observed that there is no consistent value for k fold or number of learners that consistently yields the highest accuracies for both cases. To analyze this, the k-fold that leads to the highest classification accuracy is kept constant and the number of learners is varied from 2 to 10 learners. The maximum of 10 learners was chosen for its optimal balance of accuracy, computational efficiency, and risk of overfitting. Beyond this point, no significant gains in accuracy were observed. Likewise, the number of learners that achieved the highest accuracy is kept constant and the k-fold varied from 2 to 12. A maximum k-fold value of 12 was selected as it yielded optimal accuracy, with no significant improvements observed beyond this point. The rate of change from the maximum accuracy for case 1 and case 2 is calculated and presented in Table 6.5 and Table 6.6.

From Table 6.5 it can be seen that utilizing a k fold value of 12 for both cases lead to the most consistent performance with a negligible percentage change of 0.3536 from the highest detection accuracy of 94.24% for case 1 and 92.60% for case 2. This suggests that when using a *k* fold value of 12, the accuracy of the detection model remains stable. Similarly, Table 6.6 shows that employing 10 learners also results in a minimal percentage change of 0.2560 from the best detection accuracy of 94.24% for case 1 and 92.60% for case 2.

Table 6.5: Percentage Change when Number of Learners is Constant with Kfold 2 to 12

k-fold	Case 1 Accuracy (%)	% Change	Case 2 Accuracy (%)	% Change	Total % Change
2	92.59	1.74	84.94	8.27	10.01
3	93.86	0.39	87.41	5.60	5.99
4	93.53	0.75	88.64	4.27	5.02
5	93.06	1.25	90.35	2.42	3.67
6	94.24	0	88.90	3.98	3.98
7	94.17	0.07	87.16	5.87	5.94
8	94.00	0.25	86.82	6.23	6.48
9	93.65	0.62	90.95	1.77	2.39
10	93.88	0.38	87.62	5.37	5.75
11	93.93	0.33	91.11	1.60	1.93
12	93.91	0.35	92.60	0	0.35

Table 6.6: Percentage Change when KFOLD is Constant with Number of Learners 2 to 10

Number of learners	Case 1 Accuracy (%)	% Change	Case 1 Accuracy (%)	% Change	Total % Change
2	92.12	2.25	90.81	1.89	4.14
3	92.68	1.65	92.35	0.26	1.91
4	93.30	0.99	92.60	0	0.99
5	93.78	0.48	91.56	1.11	1.59
6	93.98	0.27	91.84	0.82	1.09
7	94.02	0.23	91.92	0.72	0.95
8	94.08	0.17	92.28	0.33	0.5
9	93.99	0.26	92.18	0.44	0.7
10	94.24	0	92.36	0.25	0.25

Therefore, it can be recommended per test system 1, that using a k fold value of 12 and employing 10 learners will achieve optimal performance in both cases. These findings offer insights into how sensitive the proposed approach is to variations in the *k* fold and number of learners parameters.

6.6. Result of Implementing the Proposed Approach on the 18-IED Dataset

Table 6.7 lists the detection accuracies obtained when implementing the proposed FTBE classifier on the 18-IED dataset. An in-depth exploration of k-fold and number of learner values, both ranging from 2 to 10, was performed through a nested iterative process to obtain the highest accuracy, which was then kept constant for further analysis. Visual inspection of the results reveals that the proposed FTBE also outperforms the other machine learning classifiers as it was able to achieve an accuracy of 100% in all cases. In regard to Case 1, the classification accuracy is most effectively achieved with 3 learners and a k-fold setting of 2. In Case 2, the optimal results are obtained by employing 8 learners and a k-fold setting of 7. In Case 3, the highest accuracy is achieved by utilizing 2 learners and a k-fold setting of 4. Lastly, for Case 4, the maximum accuracy is attained with the employment of 3 learners and a fold setting of 6. This result further demonstrates the insensitivity of the proposed FTBE classification approach when selecting the datasets for training and testing. Additionally, the result also includes the precision, recall, and F1-scores. It is important to note that "NaN" values exist for the DT, KNN, and SVM machine learning classifiers. This is due to instances in which no affirmative sample is predicted for a particular scenario. Therefore, precision and F1-score values cannot be calculated for these instances. In addition, the other machine learning classifiers also had poor recall values in comparison to the proposed FTBE approach, resulting in lower F1-scores. The

reason for the low recall value in other machine learning algorithms is that the datasets from this test system are highly imbalanced. This means there is a significant difference in the number of instances between classes. Due to this imbalance, it becomes difficult for these algorithms to correctly identify cyberattacks, which are the minority class in this case. However, the proposed FTBE classifier was able to overcome this challenge and achieve high recall values resulting in higher F1 score values compared to other machine learning classifiers. This further demonstrates the effectiveness and suitability of the learning approach for dealing with class imbalance issues in cyberattack problems.

Table 6.7: Comparative Analysis of the Accuracy Results of the 18-IED Dataset

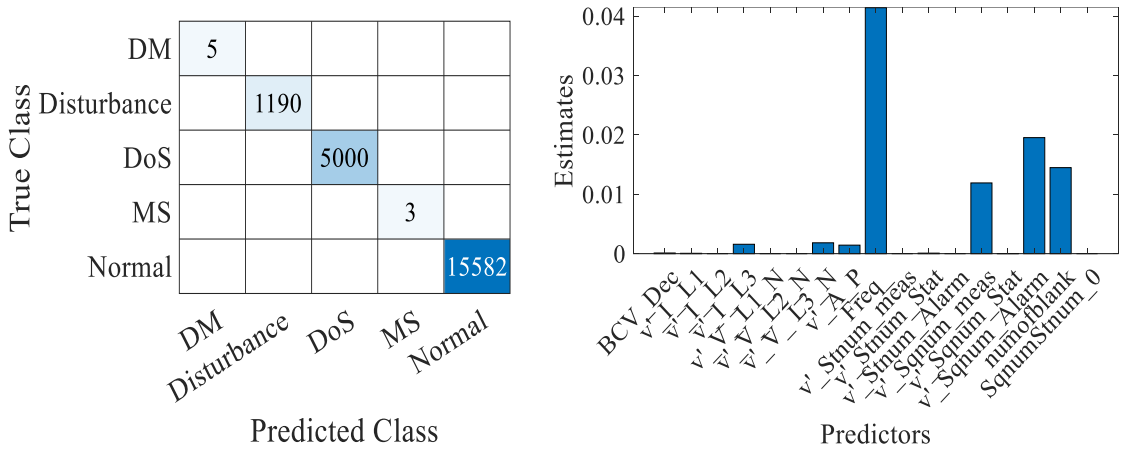
Case	Method	Accuracy (%)	Precision	Recall	F1-score
1	DT	99.99	0.9200	0.9998	0.9583
	KNN	99.84	NaN	0.5974	NaN
	SVM	99.88	NaN	0.7981	NaN
	FTBE	100	1	1	1
2	DT	99.96	0.9000	0.8857	0.8927
	KNN	99.77	NaN	0.5982	NaN
	SVM	96.44	NaN	0.6342	NaN
	FTBE	100	1	1	1
3	DT	96.71	0.5991	0.5502	0.5736
	KNN	99.79	NaN	0.5982	NaN
	SVM	96,56	NaN	0.5485	NaN
	FTBE	100	1	1	1
4	DT	99.96	NaN	0.8000	NaN
	KNN	99.83	NaN	0.5972	NaN
	SVM	99.85	NaN	0.5983	NaN
	FTBE	100	1	1	1

6.6.1. Confusion Matrices and Features Identification for Cyberattack Types

The classification of attack types through the FTBE based approach proposed herein proved successful according to the findings illustrated by confusion matrices and predictor importance plots shown in Figure 6.27 – Figure 6.30. These visual representations of the confusion matrices clearly exhibit a strong concordance between true and predicted classes underscoring accurate classification outcomes. By focusing on significant physical and network features highlighted through predictor importance plots, valuable insights are gained into factors contributing to this success. A closer analysis identified frequency (Freq) as having the highest PI values among the examined physical features. Additionally, three network features - Boolean to decimal conversion (BCV_Dec), GOOSE sequence number (Sqnum) and numofblanks - also demonstrated high PI values. These results suggest that Freq, BCV_Dec, Sqnum, and numofblanks are indispensable for successful classification of attack types of this test system data.

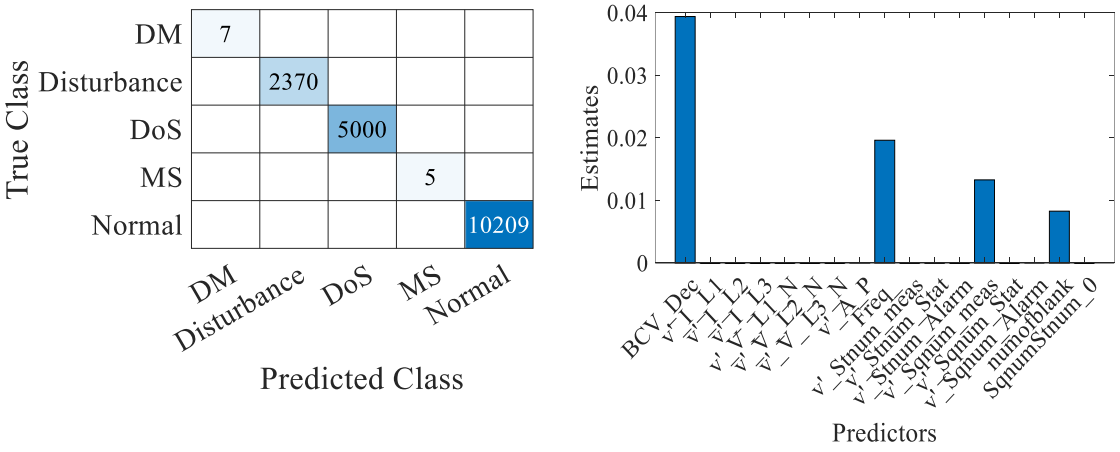
Further investigation revealed that specific features excelled at identifying attack categories. For instance, the changes in frequency (Freq) proved instrumental in detecting Data Manipulation attacks. Furthermore, the unique sequence numbers assigned to GOOSE messages, determined by the IEC 61850 protocol enabled the effective identification of Message Suppression attacks through the Sqnum feature. This feature serves as a more direct and efficient way to identify Message Suppression attacks because if a message is suppressed, a gap or inconsistency in the sequence of numbers will occur, signaling a Message Suppression attack. Also, by detecting sudden changes in circuit breaker and disconnector status, the Boolean to decimal conversion (BCV_Dec) feature successfully detected anomalies indicative of potential data manipulation. Lastly,

identifying DoS attacks was made possible by leveraging the numofblanks feature, which overwhelms system capacity with 5000 packets and results in gaps within communication logs where typical traffic would be recorded.



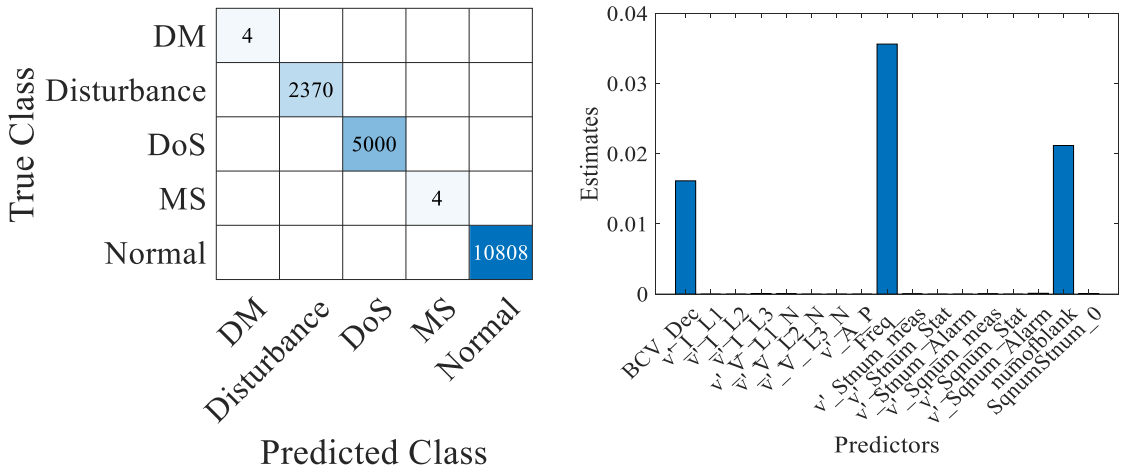
(a) (b)

Figure 6.27: Classification results of case 1. (a) Confusion Matrix and (b) predictor importance.



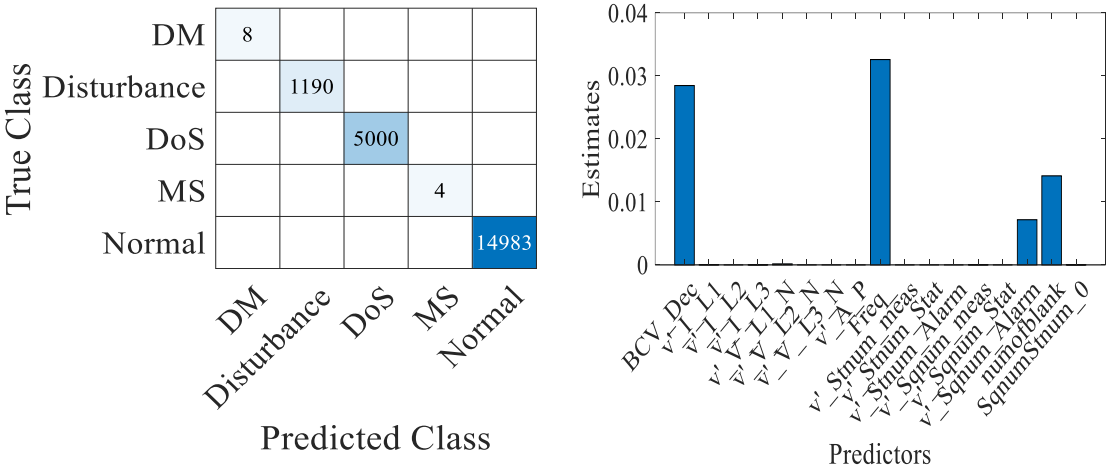
(a) (b)

Figure 6.28: Classification results of case 2. (a) Confusion Matrix and (b) predictor importance.



(a) (b)

Figure 6.29: Classification results of case 3. (a) Confusion Matrix and (b) predictor importance.



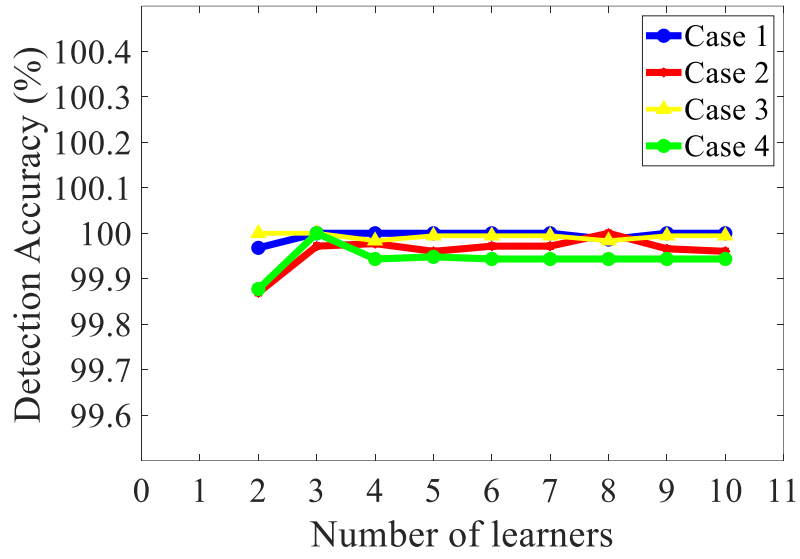
(a) (b)

Figure 6.30: Classification results of case 4. (a) Confusion Matrix and (b) predictor importance.

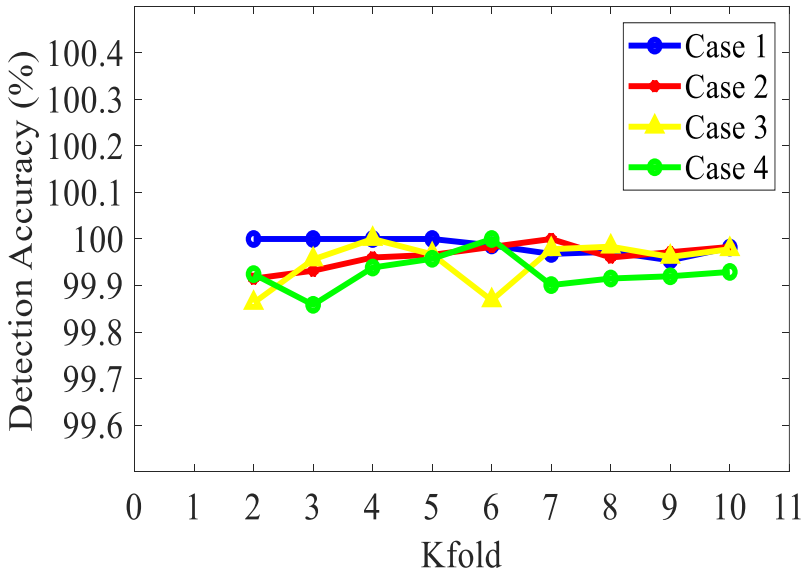
6.6.2. Sensitivity and Error Analysis

To assess the robustness of the FTBE approach a sensitivity analysis is also conducted on the 18-IED dataset. The focus is also to understand how the changes in the k-fold parameter and the number of learners may affect the results. Using brute force, the highest accuracy for each case was achieved with the following k-fold and learner combinations: Case 1 with 2-folds and 3 learners, Case 2 with 7-folds and 8 learners, Case 3 with 4-folds and 2 learners, and Case 4 with 6-folds and 3 learners. It is observed that there is no consistent value for k fold or number of learners that consistently yields the highest accuracies for all cases. By varying the number of learners from 2 to 10 and the k folds from 2 to 10, the percentage change in accuracy from the maximum classification accuracy when these parameters are varied is calculated for all cases. A maximum of 10 folds and 10 learners was chosen due to the lack of significant accuracy improvement beyond these values. Figure 6.31(a) shows the plot of the detection accuracy against the number of learners while keeping the k-fold constant. It is observed that changing the number of learners had only a minimal impact on accuracy. When using 3 learners, it can be seen that there was only a small percentage difference of 0.0284 from the maximum classification accuracy of 100%. This had the least percentage change across all cases when the k-fold is kept constant. Therefore, this suggests that employing just 3 learners is sufficient for achieving excellent results.

Figure 6.31(b) shows the plot of the detection accuracy against the k-fold while keeping the number of learners constant. Similarly, when the k-fold is adjusted it is noticed that the accuracy changes were also minor. Using a k fold value of 4 across all cases resulted in an insignificant percentage difference of 0.1012 from the maximum classification accuracy.



(a)



(b)

Figure 6.31: Classification accuracies for cases 1 to 4. (a) different number of learners and (b) different k-folds.

Based on these findings it is clear that the proposed approach remains highly stable when using a k-fold value of 4. This parameter holds great significance in the approach and ensures reliable and robust results.

6.7. Summary

In this chapter, the FTBE approach was implemented for the detection and classification of different cyberattack types. The approach was tested on two different datasets, and the relevant physical and network features were evaluated to help the model identify different cyberattack types accurately. The results indicate that the FTBE approach performs consistently without being affected by the selected training and testing data, addressing issues of overfitting and sensitivity. The performance of the proposed approach was evaluated based on detection accuracy, precision, recall, and F1-score, and its sensitivity and error analysis were assessed on both datasets. The findings revealed that the proposed approach could classify all cyberattack cases in the 18-IED dataset with 100% accuracy. Moreover, the proposed approach successfully classified 94.24% of scenarios in case 1 and 92.60% of scenarios in case 2 of the 4-IED dataset.

7. Conclusion and Recommendations

7.1. Conclusion

The work in this thesis aims to detect and classify cyberattacks in IEC 61850 protocol-based Substation Automation System. After reviewing the state-of-the-art literature, it became evident that firstly, most of the research focused only on the detection of cyberattacks without identifying the type of cyberattack the system experienced. Secondly, the classification approaches presented in the past did not evaluate their approach with different datasets to avoid the issue of overfitting. Therefore, the main contribution of the proposed approach is the capability of the model to detect and classify cyberattack types from normal and disturbance events on two different datasets and hence facilitating more precise mitigative measures.

In this research, the feasibility of using the FTBE approach to learn and classify types of cyberattacks was explored. The approach involves creating subsets of training data by selecting from the primary dataset. By employing a bagging-based ensemble strategy several decision trees were developed by finding the best split for the tree generation process. The purity of tree nodes is evaluated using the Gini index. To control the depth and avoid overfitting, a fine tree model with numerous leaves is established. These fine trees allow for classification with up to 100 splits. After the construction of these fine trees from the randomized subsets, the final classification decisions using a majority voting system is made. The performance of the proposed FTBE-based approach and other machine learning classifiers was compared. The results from the 4-IED dataset shows that classification accuracy exceeds 92% for case 1 and case 2 while on the 18 IED dataset, it achieved an accuracy of 100% across all four cases. These results demonstrate that the

approach does not only outperform other machine learning classifiers but it also remains flexible and insensitive to the selection of training and testing datasets subjected to cyberattacks. Additionally, the proposed FTBE based approach was subjected to a sensitivity analysis to examine the impact of parameter selection such as the choice of k-fold and the number of learners on classification accuracy. The results obtained from analyzing the 4-IED dataset indicated that using a k-fold value of twelve and employing ten learners yielded the highest accuracy. Conversely, when examining the outcomes, from the 18-IED dataset, it was found that a k-fold value of four and utilizing three learners provided the highest accuracy.

Furthermore, in order to find the most salient physical and network features needed to classify the cyberattack types, this thesis proposed the use of the predictor importance technique. This technique involves assigning scores to input features of the model, which helps indicate how important each feature is when making predictions. The findings demonstrate that in the 4 IED dataset, the current, as a feature stands out in identifying both FDIA and replay attacks. Similarly, among the network features, the binary state information reflecting the status of an IED in GOOSE messages proves to be the most valuable for recognizing replay attacks. In the case of the 18-IED dataset, the frequency emerges as a distinctive feature for identifying data manipulation attacks. Also, within the network features, sequence number (Sqnum) and number of blanks (numofblanks) in GOOSE messages are indicators for identifying message suppression and denial-of-service attacks.

7.2. Recommendations

Based on the work presented in this thesis, the following represents the thesis recommendations for classifying cyberattack types in smart grid. Firstly, it is crucial that researchers focus on developing methodologies that accurately classify attack types. This will help in creating targeted countermeasures to combat these attacks effectively. Secondly, it is important to validate these methodologies using several datasets to ensure their effectiveness across various cyberattack scenarios and avoid any potential issues of overfitting. Lastly, selecting the parameters for classification accuracy is essential, for conducting sensitivity analysis. In future studies, it would be highly beneficial to explore automated or semi-automated approaches that can dynamically identify optimal parameters and enhance model optimization.

7.3. Future Work

Some next steps that can be taken to build the work presented in the thesis are: Potential mitigation strategies for cyberattacks in Smart Grids. For integrity-targeted attacks like false data manipulation attacks, the use of encrypted transmissions is recommended. To combat availability-targeted attacks such as denial-of-service, techniques like rate limiting could be employed. These approaches would enhance the cybersecurity of Substation Automation Systems. Researchers can also implement this cyberattack classification approach in the field of Electric Vehicles and Vehicle-to-Everything (V2X) communication. As vehicles increasingly communicate with various entities like traffic lights, other vehicles, and city infrastructure, understanding and expanding the classification approach to V2X communication will be crucial.

References

[1] K. Kimani, V. Oduol, and K. Langat, "Cyber security challenges for IoT-based smart grid networks," *International journal of critical infrastructure protection*, vol. 25, pp. 36-49, 2019.

[2] Z. Jiang *et al.*, "A vision of smart transmission grids," in *2009 IEEE Power & Energy Society General Meeting*, 2009: IEEE, pp. 1-10.

[3] J. N. Bharothu, M. Sridhar, and R. S. Rao, "A literature survey report on Smart Grid technologies," in *2014 International Conference on Smart Electric Grid (ISEG)*, 2014: IEEE, pp. 1-8.

[4] D. V. Dollen, "Report to NIST on the smart grid interoperability standards roadmap," in "Electric Power Research Institute (EPRI)," Gaithersburg, MD, USA, 2009. Accessed: July 28, 2023. [Online]. Available: https://www.nist.gov/system/files/documents/smartgrid/Report_to_NIST_August10_2.pdf

[5] F. Wang, Z. Lei, X. Yin, Z. Li, Z. Cao, and Y. Wang, "Information security in the smart grid: Survey and challenges," in *Geo-Spatial Knowledge and Intelligence: 5th International Conference, GSKI 2017, Chiang Mai, Thailand, December 8-10, 2017, Revised Selected Papers, Part I 5*, 2018: Springer, pp. 55-66.

[6] Wikipedia. "Stuxnet." https://en.wikipedia.org/wiki/Stuxnet#:~:text=Stuxnet%20reportedly%20ruined%20almost%20one,1%2C000%20machines%20to%20physically%20degrade (accessed 29 July 2023).

[7] R. Khan, P. Maynard, K. McLaughlin, D. Laverty, and S. Sezer, "Threat analysis of blackenergy malware for synchrophasor based real-time control and monitoring in smart grid," in *4th International Symposium for ICS & SCADA Cyber Security Research 2016 4*, 2016, pp. 53-63.

[8] J. Ding, A. Qammar, Z. Zhang, A. Karim, and H. Ning, "Cyber threats to smart grids: Review, taxonomy, potential solutions, and future directions," *Energies,* vol. 15, no. 18, p. 6799, 2022.

[9] C. Harper. "First Ever DoS Cyber-Attack On A US Power Grid Detailed In Startling Report." https://hothardware.com/news/dos-us-power-grid (accessed 30 July, 2023).

[10] IronNet. "Industroyer2 Malware Targeting Ukrainian Energy Company." https://www.ironnet.com/blog/industroyer2-malware-targeting-ukrainian-energy-company (accessed 30 July 2023.

[11] M. Lapierre. "Pro-Russian group claims responsibility for cyberattack against Hydro-Québec." https://www.cbc.ca/news/canada/montreal/hydro-quebec-website-cyberattack-1.6808947 (accessed 30 July, 2023).

[12] B. Fowler. "Data breaches break record in 2021." CNET. https://www.cnet.com/news/privacy/record-number-of-data-breaches-reported-in-2021-new-report-says/ (accessed 30 July, 2023).

[13] P. Anand, Y. Singh, A. Selwal, P. K. Singh, R. A. Felseghi, and M. S. Raboaca, "Iovt: Internet of vulnerable things? threat architecture, attack surfaces, and vulnerabilities in internet of things and its applications towards smart grids," *Energies,* vol. 13, no. 18, p. 4813, 2020.

[14] V. C. Gungor *et al.*, "A survey on smart grid potential applications and communication requirements," *IEEE Transactions on industrial informatics*, vol. 9, no. 1, pp. 28-42, 2012.

[15] K. Kaneda, S. Tamura, N. Fujiyama, Y. Arata, and H. Ito, "IEC61850 based substation automation system," in *2008 Joint International Conference on Power System Technology and IEEE Power India Conference*, 2008: IEEE, pp. 1-8.

[16] R. E. Mackiewicz, "Overview of IEC 61850 and Benefits," in *2006 IEEE Power Engineering Society General Meeting*, 2006: IEEE, p. 8 pp.

[17] SmartGrid.gov, "Smart Grid Asset Descriptions," 2011. [Online]. Available: https://www.smartgrid.gov/files/documents/description_of_assets-1.pdf.

[18] P. I. Radoglou-Grammatikis and P. G. Sarigiannidis, "Securing the smart grid: A comprehensive compilation of intrusion detection and prevention systems," *Ieee Access*, vol. 7, pp. 46595-46620, 2019.

[19] C. Peng, H. Sun, M. Yang, and Y.-L. Wang, "A survey on security communication and control for smart grids under malicious cyber attacks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 8, pp. 1554-1569, 2019.

[20] A. Nourian and S. Madnick, "A systems theoretic approach to the security threats in cyber physical systems applied to stuxnet," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 1, pp. 2-13, 2015.

[21] Y. Xu, Y. Yang, T. Li, J. Ju, and Q. Wang, "Review on cyber vulnerabilities of communication protocols in industrial control systems," in *2017 IEEE Conference on Energy Internet and Energy System Integration (EI2)*, 2017: IEEE, pp. 1-6.

[22] R. Dove, "Protecting Our Energy Infrastructure from Cyber Security Exploitation," ed. Medium, 2020.

[23] L. s. a. t. U. o. C. C. f. R. Studies, "The insurance implications of a cyber attack on the US power grid," in "Innovation Series," 2020 2015. [Online]. Available: https://www.jbs.cam.ac.uk/wp-content/uploads/2020/08/crs-lloyds-business-blackout-scenario.pdf

[24] A. Anwar and A. N. Mahmood, "Cyber security of smart grid infrastructure," *arXiv preprint arXiv:1401.3936*, 2014.

[25] M. H. a. C. Danner, "Power Back On After Blackout Strikes Much of Manhattan," ed: Intelligencer, 2019.

[26] I. r. Arroyo, "The greatest blackouts in history," ed: Sacyr.

[27] E. D. Knapp and R. Samani, *Applied cyber security and the smart grid: implementing security controls into the modern power infrastructure*. Newnes, 2013.

[28] M. C. Ju-min Park. "South Korea blames North Korea for December hack on nuclear operator." Reuters. https://www.reuters.com/article/us-nuclear-southkorea-northkorea-idUSKBN0MD0GR20150317 (accessed 31 July, 2023).

[29] L. Wei, D. Gao, and C. Luo, "False data injection attacks detection with deep belief networks in smart grid," in *2018 Chinese Automation Congress (CAC)*, 2018: IEEE, pp. 2621-2625.

[30] Wikipedia. "2017 Ukraine ransomware attacks." https://en.wikipedia.org/wiki/2017_Ukraine_ransomware_attacks (accessed 31 July, 2023).

[31] I. Ilascu. "Power company Enel Group suffers Snake Ransomware attack." Bleeping Computer. https://www.bleepingcomputer.com/news/security/power-company-enel-group-suffers-snake-ransomware-attack/ (accessed 31 July, 2023).

[32] L. Abrams. "Netwalker ransomware hits Pakistan's largest private power utility." Bleeping Computer. https://www.bleepingcomputer.com/news/security/netwalker-ransomware-hits-pakistans-largest-private-power-utility/ (accessed 31 July, 2023).

[33] M. Demboski. "Industroyer2 malware targeting Ukrainian energy company." IronNet. https://www.ironnet.com/blog/industroyer2-malware-targeting-ukrainian-energy-company (accessed 31 July, 2023).

[34] H.-M. Chung, W.-T. Li, C. Yuen, W.-H. Chung, and C.-K. Wen, "Local cyber-physical attack with leveraging detection in smart grid," in *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2017: IEEE, pp. 461-466.

[35] J. Duan, W. Zeng, and M.-Y. Chow, "Resilient distributed DC optimal power flow against data integrity attack," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3543-3552, 2016.

[36] J. Sreenath, A. Meghwani, S. Chakrabarti, K. Rajawat, and S. Srivastava, "A recursive state estimation approach to mitigate false data injection attacks in power systems," in *2017 IEEE Power & Energy Society General Meeting*, 2017: IEEE, pp. 1-5.

[37] R. Xu, R. Wang, Z. Guan, L. Wu, J. Wu, and X. Du, "Achieving efficient detection against false data injection attacks in smart grid," *IEEE Access*, vol. 5, pp. 13787-13798, 2017.

[38] J. Zhao *et al.*, "Power system dynamic state estimation: Motivations, definitions, methodologies, and future work," *IEEE Transactions on Power Systems*, vol. 34, no. 4, pp. 3188-3198, 2019.

[39] M. Khalaf, A. Youssef, and E. El-Saadany, "Detection of false data injection in automatic generation control systems using Kalman filter," in *2017 IEEE Electrical Power and Energy Conference (EPEC)*, 2017: IEEE, pp. 1-6.

[40] K. Manandhar, X. Cao, F. Hu, and Y. Liu, "Detection of faults and attacks including false data injection attack in smart grid using Kalman filter," *IEEE transactions on control of network systems*, vol. 1, no. 4, pp. 370-379, 2014.

[41] M. Tariq, M. Ali, F. Naeem, and H. V. Poor, "Vulnerability assessment of 6G-enabled smart grid cyber–physical systems," *IEEE internet of things journal*, vol. 8, no. 7, pp. 5468-5475, 2020.

[42] V. K. Singh and M. Govindarasu, "Decision tree based anomaly detection for remedial action scheme in smart grid using pmu data," in *2018 IEEE Power & Energy Society General Meeting (PESGM)*, 2018: IEEE, pp. 1-5.

[43] Y. Kwon, H. K. Kim, Y. H. Lim, and J. I. Lim, "A behavior-based intrusion detection technique for smart grid infrastructure," in *2015 IEEE Eindhoven PowerTech*, 2015: IEEE, pp. 1-6.

[44] M. F. Elrawy, L. Hadjidemetriou, C. Laoudias, and M. K. Michael, "Light-weight and robust network intrusion detection for cyber-attacks in digital substations," in *2021 IEEE PES Innovative Smart Grid Technologies-Asia (ISGT Asia)*, 2021: IEEE, pp. 1-5.

[45] H. Yoo and T. Shon, "Novel approach for detecting network anomalies for substation automation based on IEC 61850," *Multimedia Tools and Applications,* vol. 74, pp. 303-318, 2015.

[46] J. Hong and C.-C. Liu, "Intelligent electronic devices with collaborative intrusion detection systems," *IEEE Transactions on Smart Grid,* vol. 10, no. 1, pp. 271-281, 2017.

[47] S. Pan, T. Morris, and U. Adhikari, "Developing a hybrid intrusion detection system using data mining for power systems," *IEEE Transactions on Smart Grid,* vol. 6, no. 6, pp. 3104-3113, 2015.

[48] S. Sengan, V. Subramaniyaswamy, V. Indragandhi, P. Velayutham, and L. Ravi, "Detection of false data cyber-attacks for the assessment of security in smart grid using deep learning," *Computers & Electrical Engineering,* vol. 93, p. 107211, 2021.

[49] M. F. Zia, U. Inayat, W. Noor, V. Pangracious, and M. Benbouzid, "Locational Detection of False Data Injection Attack in Smart Grid Based on Multilabel Machine Learning Classification Methods," in *2023 IEEE IAS Global Conference on Renewable Energy and Hydrogen Technologies (GlobConHT)*, 2023: IEEE, pp. 1-5.

[50] A. Revathi and D. Kumar, "An efficient system for anomaly detection using deep learning classifier," *Signal, Image and Video Processing,* vol. 11, pp. 291-299, 2017.

[51] M. Ozay, I. Esnaola, F. T. Y. Vural, S. R. Kulkarni, and H. V. Poor, "Machine learning methods for attack detection in the smart grid," *IEEE transactions on neural networks and learning systems,* vol. 27, no. 8, pp. 1773-1786, 2015.

[52] J. Landford *et al.*, "Fast sequence component analysis for attack detection in synchrophasor networks," *arXiv preprint arXiv:1509.05086,* 2015.

[53] T. H. Morris, B. A. Jones, R. B. Vaughn, and Y. S. Dandass, "Deterministic intrusion detection rules for MODBUS protocols," in *2013 46th Hawaii International Conference on System Sciences*, 2013: IEEE, pp. 1773-1781.

[54] H. Li, G. Liu, W. Jiang, and Y. Dai, "Designing snort rules to detect abnormal DNP3 network data," in *2015 International Conference on Control, Automation and Information Sciences (ICCAIS)*, 2015: IEEE, pp. 343-348.

[55] B. Kang, K. McLaughlin, and S. Sezer, "Towards a stateful analysis framework for smart grid network intrusion detection," in *4th International Symposium for ICS & SCADA Cyber Security Research 2016 4*, 2016, pp. 124-131.

[56] R. C. B. Hink, J. M. Beaver, M. A. Buckner, T. Morris, U. Adhikari, and S. Pan, "Machine learning for power system disturbance and cyber-attack discrimination," in *2014 7th International symposium on resilient control systems (ISRCS)*, 2014: IEEE, pp. 1-8.

[57] M. Keshk, N. Moustafa, E. Sitnikova, and G. Creech, "Privacy preservation intrusion detection technique for SCADA systems," in *2017 Military Communications and Information Systems Conference (MilCIS)*, 2017: IEEE, pp. 1-6.

[58] Y. A. Farrukh, Z. Ahmad, I. Khan, and R. M. Elavarasan, "A sequential supervised machine learning approach for cyber attack detection in a smart grid system," in *2021 North American Power Symposium (NAPS)*, 2021: IEEE, pp. 1-6.

[59] X. Wang, C. Fidge, G. Nourbakhsh, E. Foo, Z. Jadidi, and C. Li, "Anomaly Detection for Insider Attacks From Untrusted Intelligent Electronic Devices in Substation Automation Systems," *IEEE Access*, vol. 10, pp. 6629-6649, 2022.

[60] M. Massaoudi, S. S. Refaat, and H. Abu-Rub, "Intrusion detection method based on smote transformation for smart grid cybersecurity," in *2022 3rd International Conference on Smart Grid and Renewable Energy (SGRE)*, 2022: IEEE, pp. 1-6.

[61] Javatpoint. "Decision Tree Classification Algorithm." https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm (accessed 21 July, 2023).

[62] A. S. Musleh, G. Chen, and Z. Y. Dong, "A survey on the detection algorithms for false data injection attacks in smart grids," *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2218-2234, 2019.

[63] P. Kumar and A. S. Hati, "Review on machine learning algorithm based fault detection in induction motors," *Archives of Computational Methods in Engineering*, vol. 28, pp. 1929-1940, 2021.

[64] R. A. Sowah *et al.*, "Design of power distribution network fault data collector for fault detection, location and classification using machine learning," in *2018 IEEE 7th International Conference on Adaptive Science & Technology (ICAST)*, 2018: IEEE, pp. 1-8.

[65] J. M. Gillis, S. M. Alshareef, and W. G. Morsi, "Nonintrusive load monitoring using wavelet design and machine learning," *IEEE Transactions on Smart Grid,* vol. 7, no. 1, pp. 320-328, 2015.

[66] J. H. F. L. Breiman, R. Olsen and C. J. Stone, *Classification and Regression Trees*, 1st Edition ed. New York, NY, USA: Chapman & Hall/CRC, 1984.

[67] B. R. Patel and K. K. Rana, "A survey on decision tree algorithm for classification," *International Journal of Engineering Development and Research,* vol. 2, no. 1, pp. 1-5, 2014.

[68] H. Mirshekali, R. Dashti, A. Keshavarz, and H. R. Shaker, "Machine learning-based fault location for smart distribution networks equipped with micro-PMU," *Sensors,* vol. 22, no. 3, p. 945, 2022.

[69] O. T. Ibitoye, M. O. Onibonoje, and J. O. Dada, "Machine Learning Based Techniques for Fault Detection in Power Distribution Grid: A Review," in *2022 3rd International Conference on Electrical Engineering and Informatics (ICon EEI)*, 2022: IEEE, pp. 104-107.

[70] J. Han, J. Pei, and M. Kamber, "Data mining: concepts and techniques. 2011," ed: Elsevier, 1999.

[71] Javatpoint. "K-Nearest Neighbor(KNN) Algorithm for Machine Learning." https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning (accessed 22 July, 2023).

[72] H. Bhavsar and A. Ganatra, "A comparative study of training algorithms for supervised machine learning," *International Journal of Soft Computing and Engineering (IJSCE),* vol. 2, no. 4, pp. 2231-2307, 2012.

[73] V. Krishnaiah, G. Narsimha, and N. S. Chandra, "Survey of classification techniques in data mining," *International Journal of Computer Sciences and Engineering*, vol. 2, no. 9, pp. 65-74, 2014.

[74] ABB, "Special Report IEC 61850," 2010. [Online]. Available: https://library.e.abb.com/public/a56430e1e7c06fdfc12577a00043ab8b/3BSE063756_en_ABB_Review_Special_Report_IEC_61850.pdf.

[75] P. CODE, "Communication networks and systems in substations–Part 5: Communication requirements for functions and device models," *ed*, 2003.

[76] Wikipedia, "IEC 61850," 2016, vol. 14 July 2023. [Online]. Available: https://en.wikipedia.org/wiki/IEC_61850

[77] Z. Yongli, W. Dewen, W. Yan, and Z. Wenqing, "Study on interoperable exchange of IEC 61850 data model," in *2009 4th IEEE Conference on Industrial Electronics and Applications*, 2009: IEEE, pp. 2724-2728.

[78] B. Stojcevski, "Implementation of the IEC61850 international protocol for accurate fault location in overhead transmission lines," Victoria University, 2013.

[79] S. Amjadi and A. Kalam, "IEC61850 GOOSE performance in real time and challenges faced by power utilities," in *2015 IEEE Eindhoven PowerTech*, 2015: IEEE, pp. 1-6.

[80] S. Amjadizeynalhajelo and A. Kalam, "Device Isolation in IEC61850 Based Substation Protection Systems," *International Journal on Recent Technologies in Mechanical and Electrical Engineering (IJRMEE)*, pp. 43-48, 2015.

[81] J. M. Gers and E. J. Holmes, *Protection of electricity distribution networks*. IET, 2004.

[82] P. CODE, "Communication networks and systems for power utility automation–Part 7-2: Basic information and communication structure–Abstract communication service interface (ACSI)."

[83] M. Childers and M. Borrielli, "IEC 61850 substation experiences," 2012.

[84] J. Park, E. In, S. Ahn, C. Jang, and J. Chong, "IEC 61850 standard based MMS communication stack design using OOP," in *2012 26th International Conference on Advanced Information Networking and Applications Workshops*, 2012: IEEE, pp. 329-332.

[85] S. Amjadi, "Managing IEC61850 GOOSE Messaging in Multi-vendor Zone Substations," Victoria University, 2016.

[86] T. Sidhu, M. Kanabar, and P. Parikh, "Configuration and performance testing of IEC 61850 GOOSE," in *2011 International Conference on Advanced Power System Automation and Protection*, 2011, vol. 2: IEEE, pp. 1384-1389.

[87] H. Huang *et al.*, "An IEC 61850 based coordinated control architecture for a PV-storage microgrid," in *Electronics and Electrical Engineering: Proceedings of the 2014 Asia-Pacific Electronics and Electrical Engineering Conference (EEEC 2014), December 27-28, 2014, Shanghai, China*, 2015: CRC Press, p. 231.

[88] P. CODE, "Communication networks and systems for power utility automation–Part 6: Configuration description language for communication in electrical substations related to IEDs."

[89] W. Huang, "Learn IEC 61850 configuration in 30 minutes," in *2018 71st Annual Conference for Protective Relay Engineers (CPRE)*, 2018: IEEE, pp. 1-5.

[90] C. R. Ozansoy, A. Zayegh, and A. Kalam, "The application-view model of the international standard IEC 61850," *IEEE Transactions on Power Delivery*, vol. 24, no. 3, pp. 1132-1139, 2009.

[91] T. Peirelinck, A. I. Bratcu, and Y. Bésanger, "Impact of IEC 61850 GOOSE communication quality on decentralized reactive power control in smart distribution grids—A co-simulation study," in *2016 IEEE Electrical Power and Energy Conference (EPEC)*, 2016: IEEE, pp. 1-6.

[92] D. Hou and D. Dolezilek, "IEC 61850–what it can and cannot offer to traditional protection schemes," *Schweitzer Engineering Laboratories, Inc,* vol. 20080912, 2008.

[93] Y. Lopes, D. C. Muchaluat-Saade, N. C. Fernandes, and M. Z. Fortes, "Geese: A traffic generator for performance and security evaluation of IEC 61850 networks," in *2015 IEEE 24th International Symposium on Industrial Electronics (ISIE)*, 2015: IEEE, pp. 687-692.

[94] D. M. Ingram, P. Schaub, R. R. Taylor, and D. A. Campbell, "Performance analysis of IEC 61850 sampled value process bus networks," *IEEE Transactions on industrial informatics,* vol. 9, no. 3, pp. 1445-1454, 2012.

[95] A. Dagoumas, "Assessing the impact of cybersecurity attacks on power systems," *Energies,* vol. 12, no. 4, p. 725, 2019.

[96] E. Hossain, I. Khan, F. Un-Noor, S. S. Sikander, and M. S. H. Sunny, "Application of big data and machine learning in smart grid, and associated security concerns: A review," *Ieee Access,* vol. 7, pp. 13960-13988, 2019.

[97] M. A. Rahman and H. Mohsenian-Rad, "False data injection attacks with incomplete information against smart power grids," in *2012 IEEE Global Communications Conference (GLOBECOM)*, 2012: IEEE, pp. 3153-3158.

[98] Y. Z. Lun, A. D'Innocenzo, F. Smarra, I. Malavolta, and M. D. Di Benedetto, "State of the art of cyber-physical systems security: An automatic control perspective," *Journal of Systems and Software*, vol. 149, pp. 174-216, 2019.

[99] G. Liang, J. Zhao, F. Luo, S. R. Weller, and Z. Y. Dong, "A review of false data injection attacks against modern power systems," *IEEE Transactions on Smart Grid*, vol. 8, no. 4, pp. 1630-1638, 2016.

[100] D. U. Case, "Analysis of the cyber attack on the Ukrainian power grid," *Electricity Information Sharing and Analysis Center (E-ISAC)*, vol. 388, no. 1-29, p. 3, 2016.

[101] Z. Guan, N. Sun, Y. Xu, and T. Yang, "A comprehensive survey of false data injection in smart grid," *International Journal of Wireless and Mobile Computing*, vol. 8, no. 1, pp. 27-33, 2015.

[102] S. Basumallik, "A taxonomy of data attacks in power systems," *arXiv preprint arXiv:2002.11011*, 2020.

[103] A. Aribisala, M. S. Khan, and G. Husari, "Machine learning algorithms and their applications in classifying cyber-attacks on a smart grid network," in *2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2021: IEEE, pp. 0063-0069.

[104] M. T. Library. "Common Types of Network Attacks." Microsoft. http://technet.microsoft.com/en-us/library/cc959354.aspx (accessed 7 July 2023).

[105] M. T. A. Rashid, S. Yussof, Y. Yusoff, and R. Ismail, "A review of security attacks on IEC61850 substation automation system network," in *Proceedings of the 6th International Conference on Information Technology and Multimedia*, 2014: IEEE, pp. 5-10.

[106] J. Hong, C.-C. Liu, and M. Govindarasu, "Detection of cyber intrusions using network-based multicast messages for substation automation," in *ISGT 2014*, 2014: IEEE, pp. 1-5.

[107] J. Hoyos, M. Dehus, and T. X. Brown, "Exploiting the GOOSE protocol: A practical attack on cyber-infrastructure," in *2012 IEEE Globecom Workshops*, 2012: IEEE, pp. 1508-1513.

[108] A. Krigman. "Cyber Autopsy Series: Ukrainian Power Grid Attack Makes History." GlobalSign by GMO. https://www.globalsign.com/en/blog/cyber-autopsy-series-ukranian-power-grid-attack-makes-history (accessed 7 July, 2023).

[109] N. S. Kush, E. Ahmed, M. Branagan, and E. Foo, "Poisoned GOOSE: Exploiting the GOOSE protocol," in *Proceedings of the Twelfth Australasian Information Security Conference (AISC 2014)[Conferences in Research and Practice in Information Technology, Volume 149]*, 2014: Australian Computer Society, pp. 17-22.

[110] S. Malladi, J. Alves-Foss, and R. B. Heckendorn, "On preventing replay attacks on security protocols," in *Proc. International Conference on Security and Management*, 2002, vol. 6.

[111] P. P. Biswas, H. C. Tan, Q. Zhu, Y. Li, D. Mashima, and B. Chen, "A synthesized dataset for cybersecurity study of IEC 61850 based substation," in *2019 IEEE*

International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), 2019: IEEE, pp. 1-7.

[112] X. W. (kaitoray). *Dataset-GOOSE-attacks*. [Online]. Available: https://github.com/kaitoray/Dataset-GOOSE-attacks

[113] H. C. T. t. a. L. Y. (liyuan3520). *IEC61850 Security Dataset*. [Online]. Available: https://github.com/smartgridadsc/IEC61850SecurityDataset

[114] D. Zhang, X. Zhou, S. C. Leung, and J. Zheng, "Vertical bagging decision trees model for credit scoring," *Expert Systems with Applications,* vol. 37, no. 12, pp. 7838-7843, 2010.

[115] R. E. Banfield, L. O. Hall, K. W. Bowyer, and W. P. Kegelmeyer, "A comparison of decision tree ensemble creation techniques," *IEEE transactions on pattern analysis and machine intelligence,* vol. 29, no. 1, pp. 173-180, 2006.

[116] Z. Xu, G. Huang, K. Q. Weinberger, and A. X. Zheng, "Gradient boosted feature selection," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 522-531.

[117] R.-H. Li and G. G. Belford, "Instability of decision tree classification algorithms," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2002, pp. 570-575.

[118] S. B. Kotsiantis, "Decision trees: a recent overview," *Artificial Intelligence Review,* vol. 39, pp. 261-283, 2013.

[119] D. Fournier and B. Crémilleux, "A quality index for decision tree pruning," *Knowledge-Based Systems*, vol. 15, no. 1-2, pp. 37-43, 2002.

[120] C. Ferri, P. A. Flach, and J. Hernández-Orallo, "Improving the AUC of probabilistic estimation trees," in *Machine Learning: ECML 2003: 14th European Conference on Machine Learning, Cavtat-Dubrovnik, Croatia, September 22-26, 2003. Proceedings 14*, 2003: Springer, pp. 121-132.

[121] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to data mining*. Pearson Education India, 2016.

[122] Q. Ren, M. Li, and S. Han, "Tectonic discrimination of olivine in basalt using data mining techniques based on major elements: a comparative study from multiple perspectives," *Big Earth Data*, vol. 3, no. 1, pp. 8-25, 2019.

[123] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, pp. 123-140, 1996.

[124] M. Kalirane. "Ensemble Learning Methods: Bagging, Boosting and Stacking." Analytics Vidhya. https://www.analyticsvidhya.com/blog/2023/01/ensemble-learning-methods-bagging-boosting-and-stacking/ (accessed 12 July, 2023).

[125] R. Silipo. "Ensemble Models: Bagging & Boosting." Analytics Vidhya. https://medium.com/analytics-vidhya/ensemble-models-bagging-boosting-c33706db0b0b (accessed 12 July, 2023).

[126] MathWorks. "Introduction to Feature Selection." https://www.mathworks.com/help/stats/feature-selection.html (accessed 8 July, 2023).

[127] MathWorks. "Predictor Importance." https://www.mathworks.com/help/stats/compactclassificationensemble.predictorimportance.html (accessed 8 July, 2023).