



REPUBLIC OF TÜRKİYE

ALTINBAŞ UNIVERSITY

Institute of Graduate Studies

Electrical and Computer Engineering

**EMBEDDING INTRUSION DETECTION IN
DISTRIBUTED COMPUTING ARTIFICIAL
INTELLIGENCE-BASED ROUTING IN AD HOC
NETWORKS**

Zainab Ali Abbood ALMAMOORI

Doctor of Philosophy

Supervisor

Assoc. Prof. Dr. Doğu Çağdaş ATILLA

İstanbul, 2023

**EMBEDDING INTRUSION DETECTION IN DISTRIBUTED
COMPUTING ARTIFICIAL INTELLIGENCE-BASED ROUTING IN
AD HOC NETWORKS**

Zainab Ali Abboud ALMAMOORI

Electrical and Computer Engineering

Doctor of Philosophy

ALTINBAŞ UNIVERSITY

2023

The dissertation titled EMBEDDING INTRUSION DETECTION IN DISTRIBUTED COMPUTING ARTIFICIAL INTELLIGENCE-BASED ROUTING IN AD HOC NETWORKS prepared by ZAINAB ALI ABBOOD ALMAMOORI and submitted on 11/04/2023 has been accepted unanimously for the degree of PhD in Electrical and Computer Engineering.

Asst. Prof. Dr. Çağatay AYDIN

Co-Supervisor

Assoc. Prof. Dr. Doğu Çağdaş ATILLA

Supervisor

Thesis Defense Jury Members:

Assoc. Prof. Dr. Doğu Çağdaş ATILLA

Department of Electrical and
Electronics Engineering,

Altınbaş University

Assoc. Prof. Dr. Oğuz ATA

Department of Software
Engineering,

Altınbaş University

Asst. Prof. Dr. Muhammad ILYAS

Department of Computer
Engineering,

Altınbaş University

Asst. Prof. Dr. Cahit KARAKUŞ

Department of Computer
Engineering,

Istanbul Esenyurt University

Asst. Prof. Dr. Aytuğ BOYACI

Department of Computer
Engineering,

Milli Savunma University

I hereby declare that this thesis meets all format and submission requirements of a PhD Thesis.

Submission date of the thesis to the Institute of Graduate Studies: ____/____/____

I hereby declare that all information presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Zainab Ali Abboud ALMAMOORI

Signature

DEDICATION

I dedicate this research to my supervisor, who has been essential in assisting me during the entire research process, as well as to my family, who has always been supportive during my difficult moments.



PREFACE

First of all, I thank God Almighty for His blessing, which enabled me to do this project of his grace.

I'd like to express thanks to the professors and Drs. who were credited with preparing this work and my supervisor Asst. Prof. Dr Doğu Çağdaş ATİLLA and Co-supervisor, Asst. Prof. Dr Çağatay AYDIN, for their perseverance and supervision. I am grateful that they were my educational staff throughout the entire research period. Their counsel and assistance during the study have been helpful. Without their tireless aid, direction, and trust in my ability, completing this work does not have been achievable. Also, I'd like to express thanks to them for opening my mind to a new universe of (knowledge, chances, and experience and giving to me a better understanding).

Of course, I don't forget my family I'd like to especially thank my parents and my brother, Dr Hayder A. Abbood, who gave to me supported throughout my life and helped me to get a PhD degree.

Finally, my wonderful friends... A message of thanks and gratitude I send to you for always standing by my side; thank you so much for all.

ABSTRACT

EMBEDDING INTRUSION DETECTION IN DISTRIBUTED COMPUTING ARTIFICIAL INTELLIGENCE-BASED ROUTING IN AD HOC NETWORKS

ALMAMOORI, Zainab Ali Abbood

Ph.D., Electrical and Computer Engineering, Altınbaş University,

Supervisor: Assoc. Prof. Dr. Doğu Çağdaş ATILLA

Co-Supervisor: Asst. Prof. Dr. Çağatay AYDIN

Date: April / 2023

Pages: 101

The flexibility and mobility of the MANET network is a feature that sets it apart from other networking. Nevertheless, for reasons including node mobility, decentralized management, and restricted bandwidth, MANETs are extra susceptible to security threats like Denial-of-Service (DoS) attacks. When protecting MANETs from attack, encryption and authentication schemes have their limits. So, we need a system with more accuracy to keep MANETs from these attacks. Indeed, the intrusion detection system (IDS) has been implemented on these networks. But it cannot adequately protect the network because it is a monitoring system only. Intrusion detection systems (IDSs) are subordinate monitor systems for mobile ad hoc networks, since they monitor network traffic and report anything unusual. Recently, many scientists have employed deep neural networks (DNNs) to address intrusion detection concerns. This study focuses on MANET network speed, response time, and packet transfer rate for these networks by utilizing artificial neural networks (ANN). End-to-end delay, and average receiving packet (E2E and ARP) are enhanced by employing deep learning (DL) which introduces numerous algorithms, interpretations, and data analyses by artificial neural networks (ANN), which may increase the number of nodes. Hence, Deep

learning (DL) algorithms may save a packet from failure via training IDS systems and making them more adaptable to the changing environment by allowing the system to make decisions while learning about attacks in the environment. In addition, statistical and logical methods. This study used MANET to detect the black-hole attack by focusing on security standards efficiency determination and identifying these nodes and mitigating network attacks using the three algorithms presented: The CBPNN (Cascading Back Propagation Neural Network) and The FFNN (Feed Forward Neural-Network), as well as The CNN (Convolution Neural Network). These algorithms are utilized to improve the effecting of IDS systems in the networks after collecting data through a network simulator (NS 2.4) and training it with the algorithms mentioned above using MATLAB. The results appear that our model has more reasonable average receiving packet (ARP) and end-to-end (E2E) performance than other current models. Our test implementation has yielded promising results, which have been achieved from CBP, FFNN, and CNN 74%, 82% and 85%, respectively, by the time (27s, 18s, and 17s).

Keywords: ANN, ARP, Black-Hole Attack, CBPNN Algorithm, CNN Algorithm, DoS, DL, DNN, E2E, FFNN Algorithm, IDS, MANET, Security

TABLE OF CONTENTS

	<u>Pages</u>
ABSTRACT	vii
LIST OF TABLES.....	xii
LIST OF FIGURES.....	xiii
ABBREVIATIONS.....	xvi
1. INTRODUCTION.....	1
1.1 A BRIEF OVERVIEW OF INTRUSION DETECTION SYSTEMS	3
1.2 THREE MAIN CATEGORIES OF IDS APPLIED TO MANET	4
1.3 REQUIREMENT OF IDS.....	5
1.4 MANET SECURITY AND ATTACKS	7
1.5 MACHINE LEARNING BASED IDS	10
1.6 PROBLEM STATEMENT AND SOLUTION	10
1.7 AIM OF THE THESIS.....	11
1.8 ORGANIZATION OF THESIS.....	12
2. LITERATURE REVIEW.....	14
2.1 INTRODUCTION.....	14
2.2 NETWORKS INTRUSION DETECTION SYSTEMS (NIDS).....	14
2.2.1 Machine Learning Technique.....	17
2.2.2 Deep Learning Technique	18
2.2.3 MANET with IDS that are Deployed by ML and DL Techniques	20

2.3	STUDIES COMPARISON	26
3.	RESEARCH METHODOLOGY	29
3.1	INTRODUCTION.....	29
3.2	THE FIRST PART, CREATING THE DATA OF THE MANET NETWORK...	29
3.3	ROUTING PROTOCOLS TYPE REACTIVE (AODV)	30
3.3.1	Attacking Model (Blackhole Attack)	30
3.3.2	IDS-AODV Model	32
3.3.3	Details of ID-AODV and Network Monitor	33
3.3.4	Procedure of the Proposed Method (Creating Dataset).....	33
3.4	THE SECOND PART, THE PROPOSED METHOD FOR TRAINING DATA BY USING DEEP LEARNING	36
3.4.1	Algorithm	38
3.4.2	Implementation of the Algorithms	40
4.	IMPLEMENTATION AND RESULTS.....	46
4.1	INTRODUCTION.....	46
4.2	THE FIRST PART, CREATING DATA OF NETWORK WITH MONITORING THE INTRUSION DETECTION SYSTEM.....	48
4.2.1	The 1 st Scenario Based on the Number of Nodes (20, 30, 50).....	49
4.2.2	The 2 nd Scenario Depends on Different Speeds of Nodes (10, 20, 30, 40, 50).	52
4.3	SUMMARY OF THE IMPLEMENTATION AND THE RESULTS FOR THE FIRST PART	55
4.4	THE SECOND PART, THE TRAINING OF DATA FOR THE PROPOSED NETWORK BY USING DEEP LEARNING ALGORITHMS	57

4.4.1	Result for the 1st Algorithm CBPNN	58
4.4.2	Result for the 2 nd Algorithm FFNN.....	60
4.4.3	Result for the 3 rd Algorithm CNN.....	62
4.5	SUMMARY OF TRAINING AND BEST RESULT OF THE ALGORITHMS ..	64
5.	RESULTS AND DISCUSSION	70
5.1	INTRODUCTION.....	70
5.2	COMPARISON OF THE DIFFERENCE AMONG (CBPNN, FFNN AND CNN) ALGORITHMS AFTER TRAINING MANETS.....	70
5.3	COMPARISON OF THE NETWORK PERFORMANCE BEFORE AND AFTER TRAINING	72
5.4	COMPARISON BETWEEN THE PROPOSED MODEL AND OTHER STUDIES	76
5.4.1	Evaluation Metrics for Comparative Study.....	76
5.4.2	Comparative Study of Four DL Models-Based IDS System with Proposed Model.....	77
6.	CONCLUSIONS	79
	REFERENCES	82

LIST OF TABLES

	<u>Pages</u>
Table 2.1: Effect of Different Types of Attacks in Manet.....	26
Table 2.2: Types of Intrusions and Common Attacks.	27
Table 2.3: Comparison With Previous Works.....	28
Table 4.1: Specifications of the Device used to Implement the Training for Each Algorithm.	47
Table 4.2: Parameter Value of the Two Scenarios	48
Table 4.3: Average Receiving Packet in the Network With (Different Nodes).....	51
Table 4.4: Average Receiving Packet in the Network With (different speeds)	54
Table 4.5: Configurations of Ann.....	57
Table 4.6: Performance of Cbpnn Algorithm.....	59
Table 4.7: Performance of Ffnn Algorithm.....	61
Table 4.8: Performance of Cnn Algorithm.....	63
Table 4.9: Performance of the Metrics (MSE, MAE, and RMSE) for Each Algorithm.	68
Table 5.1: Comparison of the Various Effects of Accuracy on Arp and E2E Delay Based on Each Algorithm's Performance Metrics.....	71
Table 5.2: Comparison of The Cnn Algorithms With Other Studies Using KddTest-21 Dataset.	77

LIST OF FIGURES

	<u>Pages</u>
Figure 1.1: Construction of Manet Network.	2
Figure 1.2: Types of Network Routing Traffic in Manet.	2
Figure 1.3: Structure of IDS.	3
Figure 1.4: Detailed Description of Centralized, Decentralized, and Distributed Intrusion Detection Systems Architectures Based on Monitors (M) and Evaluation Units (A).....	4
Figure 1.5: Threat Actors' Potential Locations in The Network (M) Stands for The Several CIDS Monitoring Locations.	7
Figure 1.6: Manet Attacks.	9
Figure 1.7: Structure of Thesis Organization.	13
Figure 2.1: Networks IDS Systems Structured.....	16
Figure 2.2: Literature Review Scope.....	17
Figure 2.3: Main Part of the Machine Learning.....	18
Figure 2.4: General Categorization of DL.....	20
Figure 2.5: Schema for setting up Wireless Manet Networks.....	21
Figure 2.6: IDS for Information Delivery in Manets.	22
Figure 2.7: IDS Network Architecture and Data Flow in MANETs.....	23
Figure 3.1: Demonstration of Victim-Attack Mobility at Two Different Times of Link Duration.....	31
Figure 3.2: Blackhole Attacker Node Function in the Ad-Hoc Network.....	32
Figure 3.3: Structure of IDS-Aodv.....	33

Figure 3.4: Flowchart to the IDS With the Routing Protocol (AODV).	35
Figure 3.5: Flowchart Describe the Proposed Model.	39
Figure 3.6: Weight Function Curve.	40
Figure 3.7: Display of the Deep Learning (DL) Process.	41
Figure 3.8: Computational of Wight for CBPNN Algorithm.	42
Figure 3.9: Computational of Wight for FFNN Algorithm.	43
Figure 3.10: Computational of Wight for CNN Algorithm.	45
Figure 4.1: Delay between nodes in the Network (E2E Delay, different No. of Nodes). ...	50
Figure 4.2: Difference Between ARP in the Network with (Different No. of Nodes).	52
Figure 4.3: Delay Between Nodes in the Network (Different No. of Speeds).	53
Figure 4.4: Difference Between ARP in the Network With (different No. of Speeds).	53
Figure 4.5: Proposed Network After Processing Data.	58
Figure 4.6: Design of the Cbpnn Algorithm.	59
Figure 4.7: Accuracy of Detecting a Black Hole Attack of the Cbpnn Algorithm.	60
Figure 4.8: Design of the Ffnn Algorithm.	61
Figure 4.9: Accuracy of Detecting a Black Hole Attack of the Ffnn Algorithm.	62
Figure 4.10: Design of the CNN Algorithm.	63
Figure 4.11: Accuracy of Detecting a Black Hole Attack of the CNN Algorithm.	64
Figure 4.12: Results of the Three Algorithms Cbpnn, FFNN and CNN (Matlab Results). 65	
Figure 4.13: Result for the Best Algorithm CNN (Matlab Results).	66
Figure 4.14: Best Validation Goal for the Performance of the Algorithm (Matlab Result).67	

Figure 4.15: Detailed Training Tape with the Design for the Best Model (Matlab Result of CNN Algorithm).....	67
Figure 4.16: Differences Among Three Algorithms to Detect Manet Attack.	69
Figure 5.1: Manet Network Performance After Training Data by Each Algorithm.....	72
Figure 5.2: Difference Between E2E Before and After Training by Proposed Algorithms.	73
Figure 5.3: Difference Between ARP before and After Training by Proposed Algorithms.	74
Figure 5.4: Performance Metric of the Proposed Model CNN With Other Studies to Detection of the Attack.....	78

ABBREVIATIONS

MANETs	: Ad-hoc Wireless Networks
WSNs	: Wireless Sensor Networks
AODV	: Ad hoc On-Demand Distance Vector Routing
IDS	: Intrusion Detection System
CIDSs	: Collaborative Intrusion Detection Systems
IDMEF	: Intrusion Detection Message Exchange Format
NIDS-DL	: Node Intrusion Detection Systems- Deep Learning
IDAODV	: Intrusion Detection Ad hoc On-Demand Distance Vector Routing
E2E	: End-to-End
ARP	: Average Received Packets
DoS	: Denial of Service
ML	: Machine Learning
AI	: Artificial Intelligence
DL	: Deep Learning
DNN-IDS	: Deep Neural Networks - Intrusion Detection System
DNN	: Deep Neural Networks
CBPNN	: Cascading Back Propagation Neural Network
FFNN	: Feedforward-Neural-Network
CNN	: Convolutional-Neural-Network
MSE	: Mean Squared Error
MAE	: Mean Absolute Error
RMSE	: Root Mean Squared Error
SL / DL	: Supervised Learning / Deep Learning

1. INTRODUCTION

Decentralized networks are gaining popularity, especially the types of networks are recognized (Mobile Ad-hoc Networks). Since it doesn't require a centralized infrastructure like router and/or gateway (in traditional networks), hosts can travel freely throughout the network without being attached to a cable. This may result in the network's topology being unstable and continually changing. MANET relies on peer-to-peer (P2P) routing with multiple intermediary nodes as opposed to centralized network nodes. When nodes aren't in direct radio communication, multi-hop routing must be implemented. Each infected host serves as an access point (the network's structure is dynamic), as nodes can readily join or depart. Represent node to each one as a router and send data from the other nodes to itself that means (there aren't any central routers). This type of network is particularly vulnerable to Denial-of-Service (DoS) assaults due to the complexity and sensitivity of its route discovery procedure, resulting from its dynamic nature [1]. Maintaining secure channels of communication is a pressing concern in the modern era. Modern security practices employ multiple layers of defense when protecting a network from a malicious node. A MANET must contain encrypted messages because of the network's inherent fragility [2] since they don't necessitate the construction of a network backbone; they can be applied in a wide collection of fields, like the military, rescue operations, and healthcare.

Moreover, the security of this network is essential in critical situations, such as wartime communications. Due to its security vulnerabilities, the network is an enticing target for hackers. It is vital to create a security system that is both efficient and adaptable in order to avoid any malicious behaviors on the network. Because the dynamic nature of nodes, it is complicated to implement safety measures in a MANET, as its topology is continually shifting. A network is self-configured as well as self-deployed, with no central communication authority [3].

Using a wireless channel to transmit data between nodes makes the network more sensitive to faults and security concerns. In multi-hop communication, numerous nodes participate in the transmission of a data packet. As a result, every node is connected to every other node, as depicted in Figure 1.1. The net architecture frequently changes, and Mobile ad-hoc nodes switch arbitrarily. MANET can be either autonomous or else integrated into a wider net.

Therefore, it establishes an extraordinarily complicated architecture where the MANET's most important, challenging aspect is equipping every node to maintain accurate traffic continuously [4]. Figure 1.2 illustrates the types of network routing traffic in Mobile Ad hoc network.

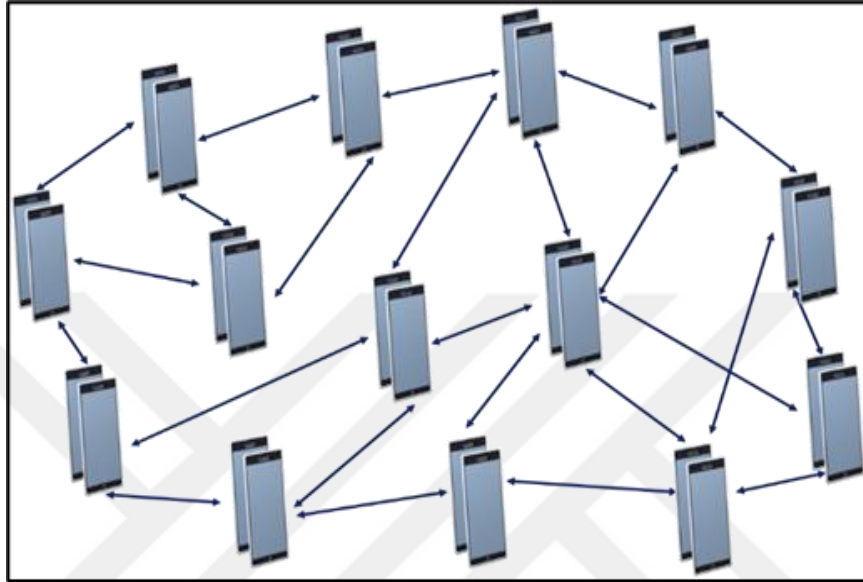


Figure 1.1: Construction of Manet Network.

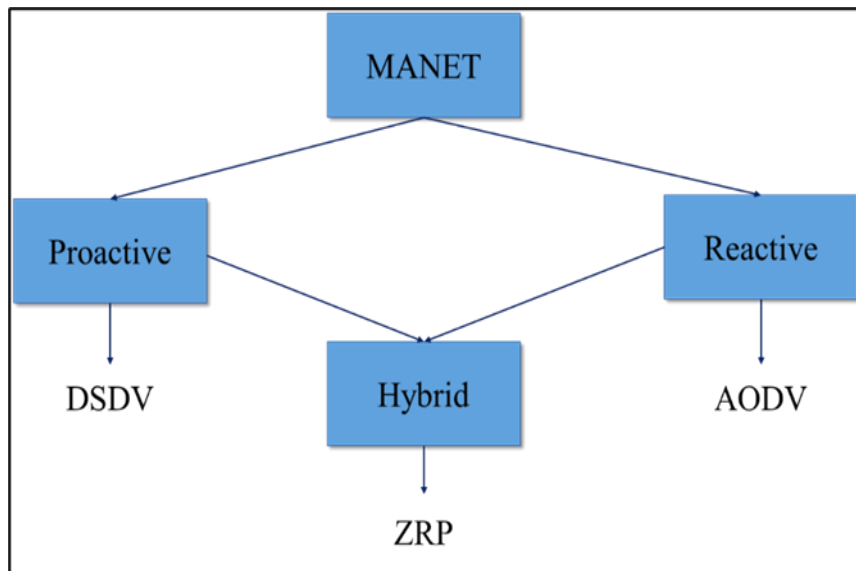


Figure 1.2: Types of Network Routing Traffic in Manet [4].

1.1 A BRIEF OVERVIEW OF INTRUSION DETECTION SYSTEMS

Anderson J. P. proposed the concept of an intrusion detection system (IDS) in 1980. According to him, an intrusion or threat is when an attempt is made to access or operate the system in a way that causes it to be unreliable [5]. In other words, an intrusion detection concept based on an operating system audit log. However, academics have shown little interest in this approach, focusing instead on encryption as well as the refusal of data access after a validated host [6]. Denning, D. E suggested intrusion detection in 1985, which includes the entity, host, Rolodex entry, outlier log, audit log, and rules of engagement [7], but this system doesn't care about OS, software, security holes, or attack vectors. In 1988, Lunt T. enhanced it by developing an intrusion detection system [8] that could identify attacks in real-time as they were being received. Invasion patterns for a given host can be predicted using the Lunt model. In 1989, Heberlein LT provided peer-to-peer intrusion detection and introduced the network security monitor, which detects unusual behaviour by analyzing network traffic in LANs rather than checking the host's audit logs. [9]. Figure 1.3 depicts the intrusion detection system's Structure (IDS).

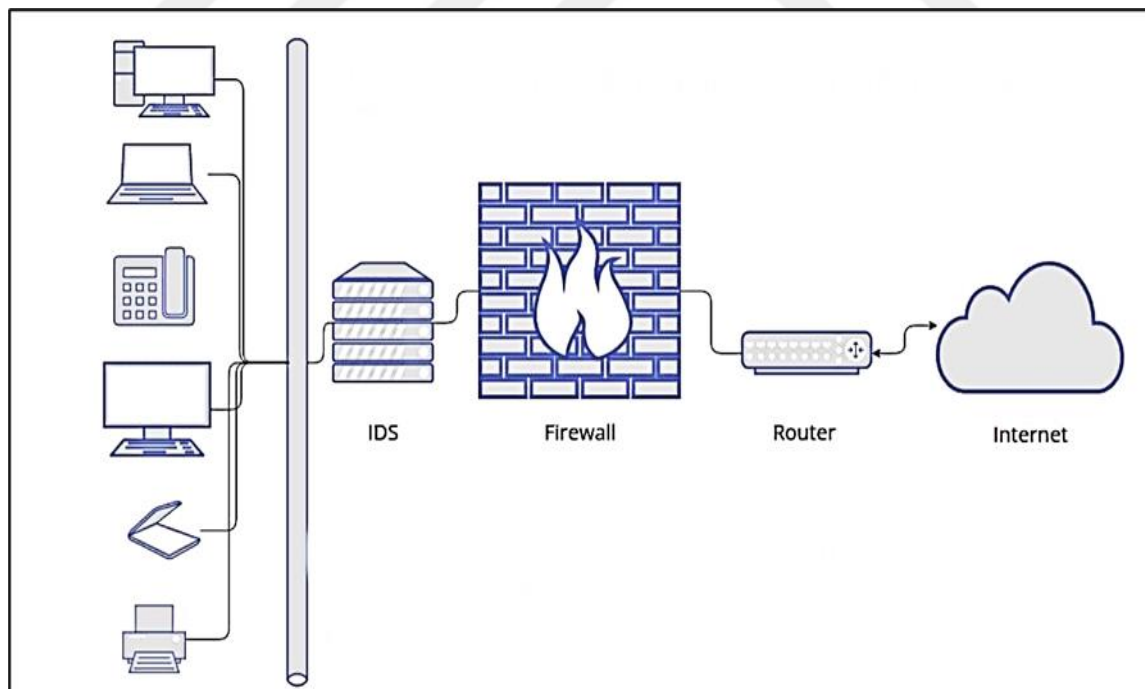


Figure 1.3: Structure of IDS [9].

1.2 THREE MAIN CATEGORIES OF IDS APPLIED TO MANET

Most of the earliest IDSs unique single instances monitored the single system or net via conducting a regional evaluation for assaults. Therefore, there is no communication and interaction amongst models of such a standalone IDS. Such a method is incapable of detecting complex and widely disseminated threats. Isolated intrusion detection will be unable to make up links between harmful events co-occurring in different locations. Consequently, Collaborative Intrusion Detection Systems (CIDSs) evolved [10] for the protection of extensive networks as well as IT ecosystems. Collaborative Intrusion Detection Systems involve multiple sensors and monitors that collect data. In addition, they often include several evaluation units that do the real IDS using sensor data. Collaborative Intrusion Detection Systems can significantly minimize the complexity of security administrators' jobs [11], boosting the accuracy of network monitoring detection. Depending on the Collaborative Intrusion Detection Systems, monitors and evaluations may also be co-situated. Collaborative Intrusion Detection Systems enforce cooperation between various monitors and are therefore more mountable than Intrusion Detection Systems that operate independently. According to their communication architecture, Collaborative Intrusion Detection Systems can be characterized as centralized, decentralized, or distributed [12]-[14], as shown in Figure 1.4.

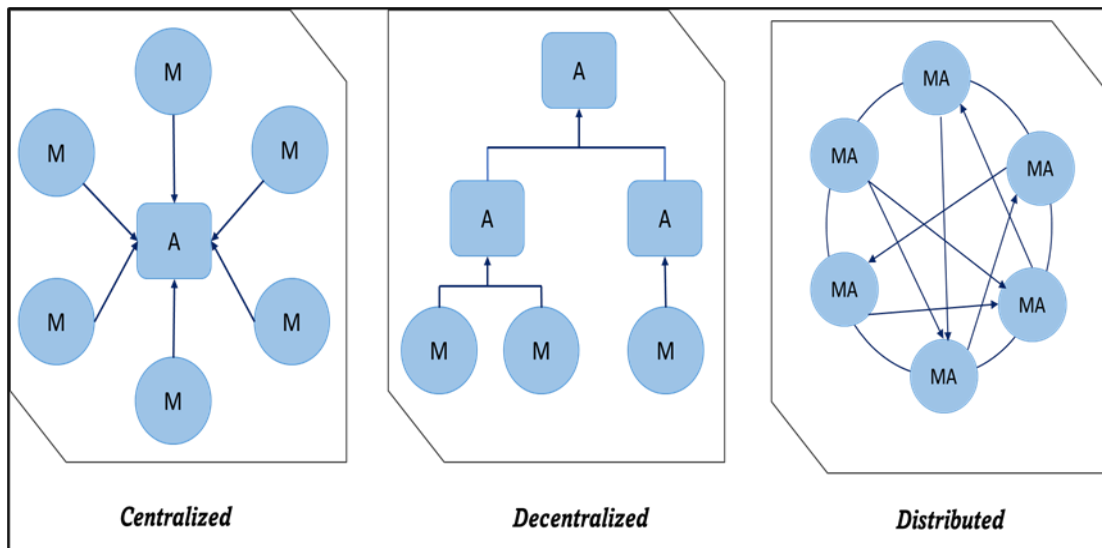


Figure 1.4: Detailed Description of Centralized, Decentralized, and Distributed Intrusion Detection Systems Architectures Based on Monitors (M) and Evaluation Units (A) [12].

- a. Centralized collaborative intrusion detection systems: use many monitors, each of which monitors its host's behaviour and traffic across the network. Data from these monitors, such as alerts generated by a local detection or information gathered from local network traffic, is shared with this centralized analysis node. This means that the analysis section either uses alert correlation techniques to process incoming alerts or applies standard detection methods to incoming network traffic data. Comparable to standalone IDSs, centralized collaborative IDSs struggle to keep up as the size of the protected system expands. Hence, the central processing unit's lack of redundancy makes it vulnerable to breakdown.
- b. Decentralized collaborative intrusion detection systems: often feature a hierarchical monitoring point topology or numerous self-controlled intrusion detection systems uses. They circumvent the performing issue of centralized collaborative intrusion detection systems by employing preprocessing, and a correlation of observed data within the hierarchy node; the data congregate on a central evaluation node at the highest.
- c. Distributed collaborative intrusion detection systems: divide the duties of a central evaluation node evenly among all displays, making each examine an evaluation node. Consequently, distributed collaborative intrusion detection systems often adopt the peer-to-peer (P2P) architecture, wherein preceding sections are correlated, aggregated, and analyzed entirely among the monitors.

1.3 REQUIREMENT OF IDS

A CIDS for safeguarding extensive networks or large IT systems must meet the following conditions [15]:

- a. Accuracy: based on the ratio of successfully identified assaults to missed ones. When determining the accuracy of an IDS, false positives (FP) must be considered where a precise IDS ought to decrease it.
- b. Minimal overhead: In terms of computing and communication, overhead arises. The procedures utilized to generate, collect, and correlate intrusion warnings must have little computing overhead. In addition, signalling within intrusion detection (for

example, among monitoring or else among monitors and evaluation) must be kept to a minimum.

- c. Scalability: It necessitates that the intrusion detection system's effectiveness improves linearly as well as the number of extra resources, enabling the protection of networks of any scale [16]. Consequently, the intrusion detection system must not include bottlenecks like a single point of failure (SPOF).
 - d. Resilience: A collaborative intrusion detection system should maintain appropriate availability and accuracy during internal component failures and attacks. Consequently, in addition to being resistant to system failures or external attacks like Denial of Service, a collaborative intrusion detection system must also be immune to internal attacks from hostile elements and systems inside the protected network or system. Therefore, collaborative intrusion detection systems must guard against (SPOF) and degradation or rapid restoration in the event of failures or attacks.
 - e. Privacy: In a cooperative approach, alerts may contain sensitive information that requires protection and cannot be shared or released with all collaborative intrusion detection systems components; This is crucial for collaborative intrusion detection systems installations that share data between domains and need privacy for the consumers, corporations, and network providers involved.
 - f. Self-organization: entails a system's capacity to self-adjust without an administrator's involvement; in contrast to systems requiring manual configuration, this enables the construction of less-prone-to-error systems.
 - g. Interoperability: is the system's capacity to communicate in conjunction with instances of the same system employed on other nets as well as various intrusion detection systems implementations; This can be accomplished, for example, by using standardized forms like the Intrusion Detection Message Exchange Format (IDMEF).
- Figure 1.5. Displays potential network positions of potential attackers, (M) represents the various collaborative IDSs monitoring points.

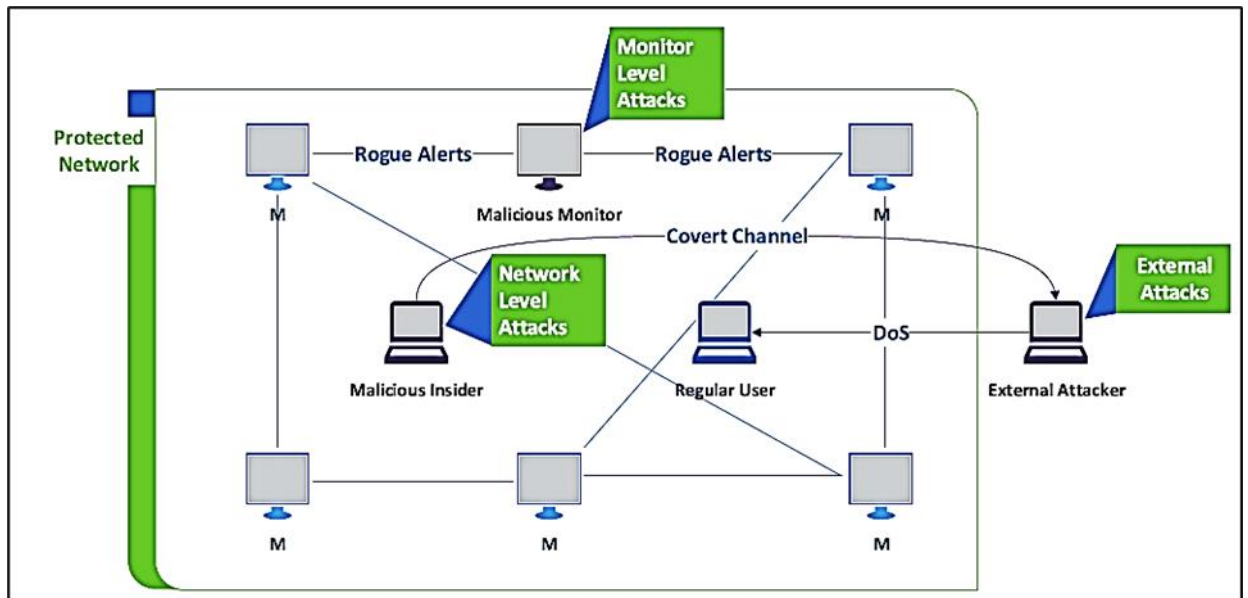


Figure 1.5: Threat actors' potential locations in the network (M) Stands for the Several CIDS Monitoring Locations [13].

1.4 MANET SECURITY AND ATTACKS

Many different types of attacks are made against MANETs due to their decentralized nature, self-management, etc. In the past decade, numerous security measures have been introduced to detect and counteract dangers. When MANETs first emerged, their security protocols were built around cryptographic mechanisms. As early as 1999, researchers were able to use threshold cryptography to evolve a key administration mechanism designed exclusively for procuring MANET. Some nodes in the network architecture were operated as hosts, in constant, others performed administrative tasks. In addition, SAODV, an upgraded variant of AODV, has been proposed [17]. It was suggested that digital signatures and hash chains might provide cryptographic security for ad hoc networks. For authentication, numerous cryptographic schemes require securities issued by a central authority [18]. Other techniques [19], including the PGP, that trust model and other innovations have altered the notion of central supervision; as part of the authentication process, a security chain is kept at the end of each node per these techniques.

In addition, the tests reveal that cryptographic techniques impose substantial communication latency and need a pre-existing link between nodes, making them inappropriate for ad hoc networks. This led to the development of a large array of hybrid strategies for increasing

MANET security. Alternatively, MANET nodes are susceptible to both active and passive attacks. There are numerous attacks, but Denial of Service (DoS) attacks, Man in the Middle (MITM) attacks, and flooding attacks from black holes and gray holes are prevalent. Several cryptographic protocols have formed the backbone of MANET security for decades. In pursuit of the most advanced, effective, and efficient MANET security solutions, academics have turned to cutting-edge technologies such as ML, DL, and AI throughout the past decade (as will be explained in the next chapter). Several of their detection, prevention, prediction, and mitigation efforts [20] depended on machine learning. Some forms of assaults to which MANET networks are susceptible are outlined below:

- a. Denial of Service Attack: An intruder's DoS attack [21] involves an injection of packets into networks to consume network resources. For instance, a hostile node can quickly seize bandwidth in a MANET if it spams the network with route request packets.
- b. Flooding Attack: A hostile node may conduct a denial-of-service (DoS) attack known as flooding to overload a network and deplete its resources. In general, flooding assault is possible with on-demand routing methods.
- c. Routing Table Runoff: Misbehaving nodes may attempt to compromise other nodes' routing databases by sending route request packets that incorrectly query for nonexistent nodes. Sometimes, fake networks would update their routing tables after receiving request packet packets from malicious networks. Given the restricted amount of RAM available, the targeted nodes' routing tables will be deleted.
- d. Impersonation Attacks: A node could pretend to be another node in order to send spoofed routing data. Additionally, attacks nodes may get unauthorized access to system, insights, and potentially issue erroneous commands or status updates to other nodes.
- e. Power Consumption Attacks: In MANET networks, the power demands of mobile nodes are of paramount importance. A malicious node that can generate enough packet traffic to attack other nodes is possible. These nodes may then are asked to pass on the packets or store the routes they took. As a result, mobile nodes lose power by sending these packets. Figure 1.6 demonstrates the attacks on the MANET depending on the network layers.

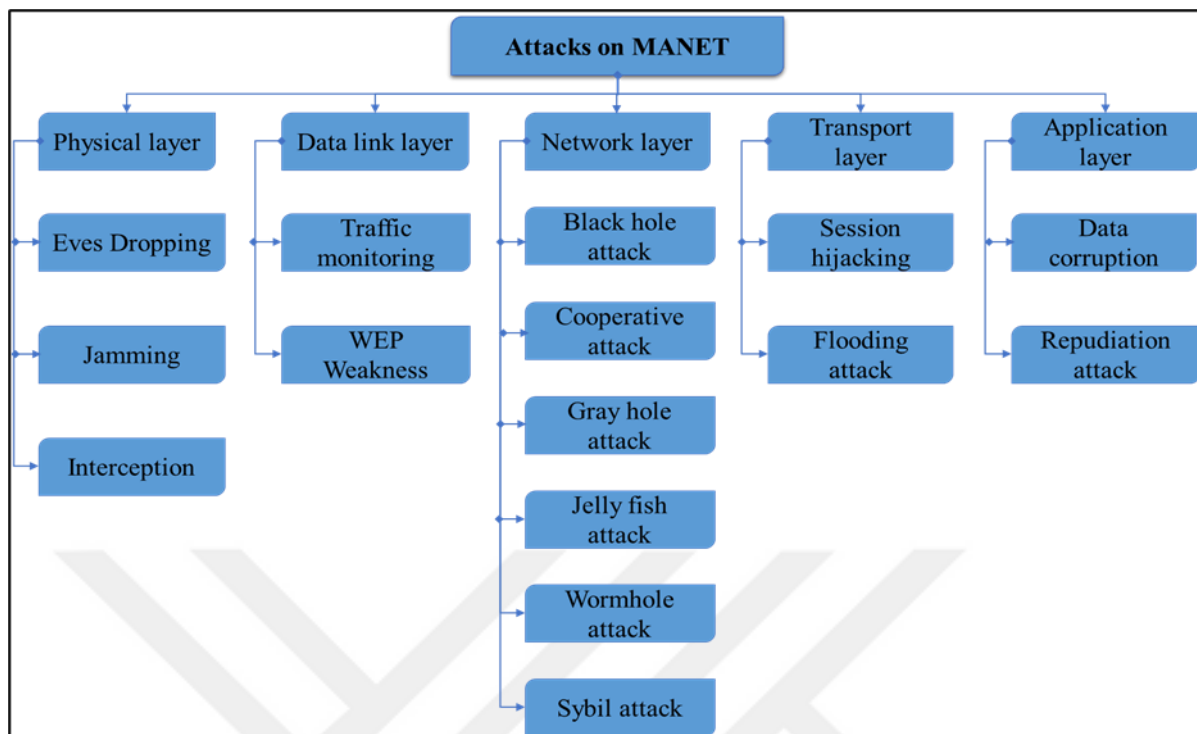


Figure 1.6: Manet Attacks [17].

Due to the numerous attacks that networks are exposed to, we conclude several facts harmful to the network and must focus on them and solve part or most of them:

- a. No single authority can accept responsibility for a MANET's security threats. Therefore, mobile ad-hoc network dispersed architecture requires safety measures.
- b. Nodes in mobile ad-hoc networks could travel or quit and join effortlessly. MANETs' frequent topology modifications are due to their adaptable nodes. It's hard to spot hungry or aggressive system gadgets.
- c. MANETs limit the capacity for sharing wireless broadcast channels so that enemies can interfere with congestion or network traffic, ad-hoc nodes have limited energy consumption, and safe solutions require extra resources for encoding and safety procedures.
- d. Mobile ad-hoc networks need physical protection. Attackers can corrupt or destroy mobile nodes in hostile conditions, making it harder to protect against an inside attack.

1.5 MACHINE LEARNING BASED IDS

One of the greatest obstacles of running MANETs is a difficulty of detecting and mitigating attacks of varying forms due to the topology of MANETs being more susceptible to attack because of the network's open, wireless design. Therefore, MANETs have an intrusion detection system that detects and examines suspicious events; also, multiple methods exist for identifying network faults. MANETs allow nodes to join and leave simultaneously, exposing the net to several threats. Therefore, the fundamental purpose of an IDS system is to prevent nasty actions from harming the network. In this system, each node is equipped with an IDS as a security measure against intruders. The nodes' low resource availability challenges real-world IDS technology using techniques based on machine learning, and numerous new threats and vulnerabilities may be identified [22]. The intrusion detection system can be customized using ML techniques like fuzzy logic, genetics, Bayesian theory, and neural networks. Various methods, such as anomaly detection, misuse detection, and signature-based detection, comprise an intrusion detection system. It is possible to discover unlawful or aberrant nodes using anomaly detection systems that compare the activity of nodes to their typical, regular patterns. An intrusion occurs when an atypical node behavior is identified. Because they rely on previously recorded signatures or behavior patterns, signature-based or abuse detection systems cannot be utilized to identify fresh assaults. Due to the fact that an attacker never follows a predetermined attack approach, outlier detection systems excel at identifying unanticipated exposures but also generate more false positives. Therefore, using machine learning based on intrusion detection systems has become an enticing selection for deploying many techniques that minimize (true negatives (TN)) as well as (false negatives (FN)). In addition to improving MANET safety. Classification methods based on ML can be used to differentiate between regular as well as intruder nodes. New intrusion detection systems (IDSs) with hierarchical and distributed architectures were introduced [23], [24].

1.6 PROBLEM STATEMENT AND SOLUTION

With the increasing demand for MANET networks as a result of their numerous benefits, the expanding number of nodes that may be employed in these networks, and their capacity for high data transmission rates. The numeral of packets communicated to exchange the

routing in these networks needs more information; the number of resources required by each node varies depending on the routing approach. To address the limitations of these issues and combine their benefits, recent methods use a distributed-computation approach, in which the network information is processed by the software defined network (SDN), as they are common across the network. The packet information is processed at the nodes during the routing using such an approach, as the nodes are only updated when a significant change occurs in the network; instead of requesting route information from the SDN per each routing task, and reduce the resources consumption, as redundant computations are eliminated because this information is processed in the SDN and delivered to all the nodes. Each node is processing only the packet information, but, despite the significant performance improvement, it became the need for IDS system of high potential to detect attacks executed against the network to increase efficiency. When an attack packet is detected and dropped earlier, rather than being sent to the next hop, significant improvement can be achieved to the efficiency of the network by eliminating the resources required to process these packets. As a result, the AI algorithm by machine learning (ML) will use to train the intrusion detection system that uses machine learning routing in MANET networks, according to the capability of these algorithms to approximate complex functions. To address the limitations of attacks against the MANETs and enhance and combine their benefits. In addition, the efficiency of the solutions used by machine learning is evaluated to increase the security of MANET in several important points:

- a. Packets and routing protocols must be secure for a network to function effectively.
- b. Sensitive applications must account for the important security aspects of the network.
- c. The accuracy of the DL model is measured via its capacity to recognize new attack forms.
- d. All the nodes in a network need to work together to determine which nodes may be trusted is therefore essential for network security, as data should not be sent to the malicious nodes.

1.7 AIM OF THE THESIS

As both the current routing methods in ad hoc networks and the intrusion detection methods rely on AI to accomplish their tasks, This study goals to improve the network's performance by combining an intrusion detection system (IDS) with a routing task and trained it by using artificial neural network (ANN). In order to detect attacks that vulnerable of MANET network. The achievement of such a goal provides the network with the following benefits:

- a. Reducing the computations required for tasks, as embedding the intrusion detection in the routing, allows the elimination of redundant computations by the neural networks used to accomplish tasks.
- b. Enhancing the effectiveness of the network by reducing the amount of recourses required to process each data in order to recognize the next hop to be sent to and reduce the number of packets that the nodes need to exchange in the MANET to route the data.
- c. Enhancing the MANET performance by reducing the link duration of the packets, as routes are discovered faster, and attack data are dropped earlier.
- d. Increase the network's average received packets (ARP) by reducing the bandwidth required to communicate routing-related data.
- e. Increasing the network speed by decreasing the end-to-end (E2E) and optimizing the routing procedure, i.e., selecting better routes for the packets to reach their destination.

1.8 ORGANIZATION OF THESIS

This thesis's remaining sections are structured as follows:

Chapter 2: Reviews the literature on ML and the techniques proposed to employ it in IDS.

Chapter 3: Explain the proposed techniques and the several kinds of ANN that be utilized to accomplish the required task.

Chapter 4: Explain the implementation and result for selected ANN to estimate the proposed model's performance.

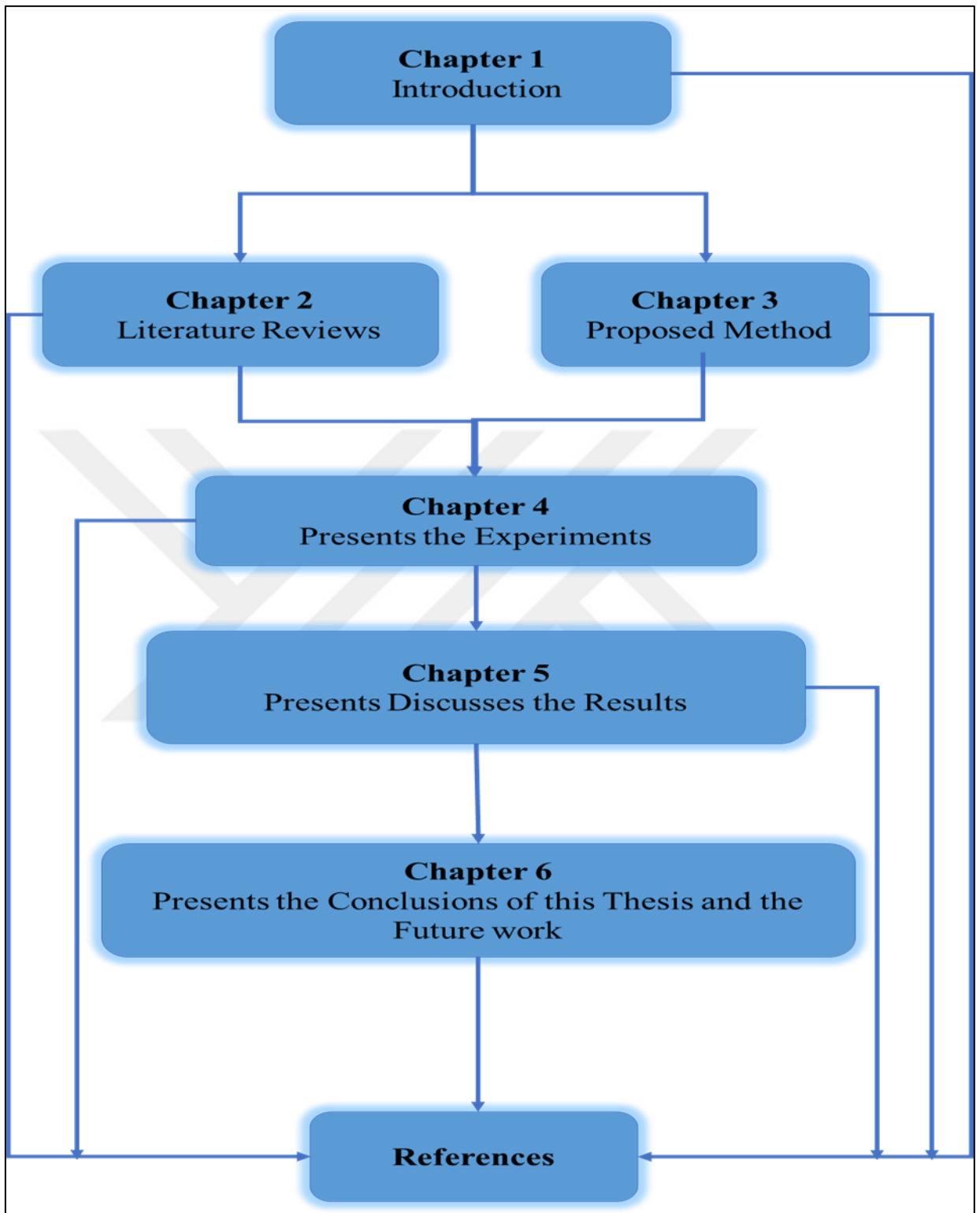


Figure 1.7: Structure of Thesis Organization.

2. LITERATURE REVIEW

2.1 INTRODUCTION

Successful attacks on computer systems often result in a direct detrimental effect on the economy. There are a variety of adversaries, including amateur enthusiasts, organized crooks, and organizations sponsored by the state. As a result of the increasing availability, the effect of internet technologies, and the rise in network threats, network defense has emerged as a significant area of study [25]. The protection of a network needs a deep knowledge of all protocols traversing the network. Therefore, taking software must incorporate numerous complicated parsers in order to interpret the acquired data correctly. Keeping up with the ever-increasing volume of data that must be processed can be difficult and calls for efficient implementations and methods. Real-time data processing is needed to detect an ongoing or past attack and limit economic damage (Zero-day) attacks to exploit daily vulnerabilities; signature-based detection cannot detect unidentified MANET threats [26].

Machine learning has been a significant study area for two decades, yet it's rarely used in software. MANET anomaly-based intrusion detection is plagued by false positives and a lack of training and analysis data [27]. Both network traffic and software stacks are changeable. To reflect, anomaly-based intrusion detection must be assessed using recent datasets. NIDS is important for finding net breaches, tracing them to their offenders, and isolating and correcting damage. The attack recognition and event tracking capabilities of IDSs have a deterrent impact on attackers, who are more likely to be caught and penalized. The presence of the IDS may induce an attacker to look for an easier-to-penetrate target [28]. Being blocked via the monitor or an analyst because of an alert draws unwelcome attention to the intruder as well as slows down his MANET operations. In order to benefit from IDS, a dependable and broad data source is required to make correct predictions.

2.2 NETWORKS INTRUSION DETECTION SYSTEMS (NIDS)

Because of the Importance of the pervasive security WSNs, it is necessary to recognize MANET risks and detail current defense plans. Traditional protection measures for known assaults have a wide range of applications and may be effective in certain circumstances;

nonetheless, they might not be entirely secure. Despite the availability of standard security measures such as authentication, access control, encryption and data secrecy, network attacks have persisted, needing a second defense line [29]. In such circumstances, the significance of IDS for MANET becomes pertinent. IDSs or NIDSs for connected networks are one of the most common security measures. Since the 1980s, NIDSs has been scrutinized to ensure the security of traditional computer science systems; Figure 2.1 shows the NIDS structure.

NIDS is, therefore, a mature science discipline. Sadly, standard NIDS techniques could work ineffectively and/or unsuitably for networks because of factors like restricted resources, energy, heterogeneity, and connection [30]. Controller nodes in conventional systems are often vital regarding computational resources and storage capacity. These nodes monitor incoming and outgoing traffic without critical asset or network bandwidth limitations. Nevertheless, MANET is distributed and composed of many devices with inadequate computing energy, storage space, and battery life.

Additionally, its network bandwidth capability is a limitation, and MANET enables unanticipated interactions between the virtual and physical worlds. Each node takes a unique IP address to ensure Internet connectivity [31]. As a result, trust issues and vulnerabilities are created. MANET is built on inhomogeneity in the range of communication protocols between coexisting technologies. Lastly, MANET environments generate vast quantities of vital data which must be protected. Consequently, it was more complex and restrictive for NIDSs in networks than in older computing systems.

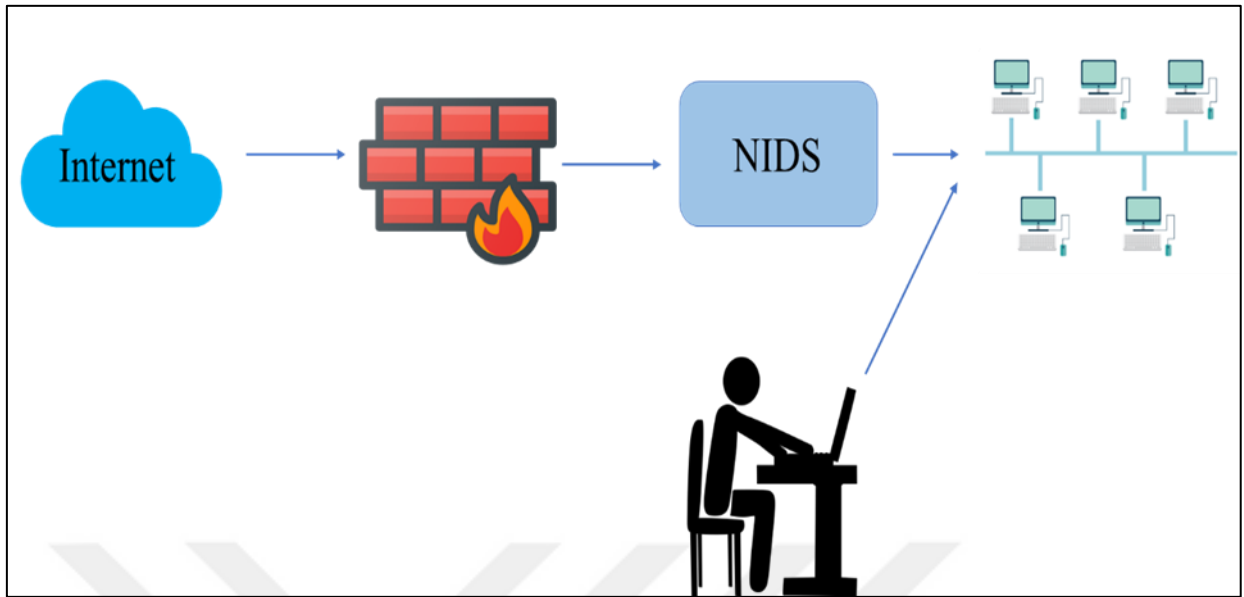


Figure 2.1: Networks IDS Systems Structured.

Numerous NIDS was created utilizing attack rules/signature or usual behaviour definition. Unfortunately, these NIDS have.

- a. Detecting unknown or zero-day attacks is a weakness.
- b. High rate of incorrect assault detectors (false alarms).

To enhance network security, researchers investigated artificial intelligence (AI) as well as machine learning (ML), with more attention to techniques of deep learning (DL) [32]-[34]. ML techniques substantially affect fraud detection, data recognition, and text categorization. The efficacy of ML has encouraged researchers to enhance the detection of attacks, anomaly detection, and the identification of abnormal behaviours within MANET networking by implementing learning algorithms into IDS. Therefore, this chapter analyzes and evaluates significant contributions to machine learning for MANETs with IDSs. In recent years, academics and industry have moved their focus to develop NIDSs based on ML. In summary, this chapter will focus on the following three crucial areas, as shown in Figure 2.2.

- a. MANET,
- b. Security mechanisms; and
- c. Machine learning techniques.

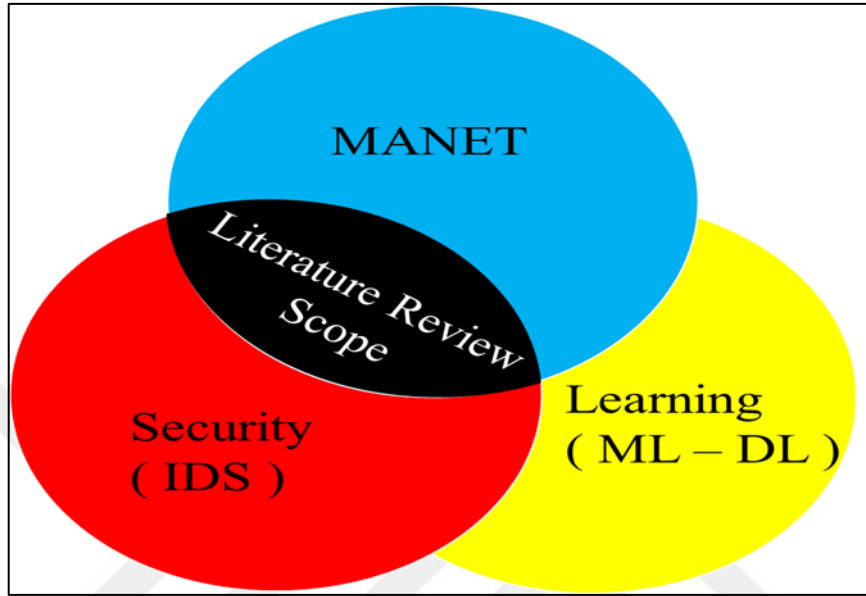


Figure 2.2: Literature Review Scope.

As seen in Figure. 2.2, we will concentrate on the overlap between the three categories.

- a. Security in the intersection point of MANET (IDS for MANET)
- b. The intersection of these three fields (MANET with IDS that are deployed via ML technique and DL technique).

2.2.1 Machine Learning Technique

Machine learning is a branch of AI technology consisting of three main parts that automatically allow machines to learn valuable data collected from large databases, as shown in Figure 2.3. Using intelligence approaches with network devices and networks solves numerous security issues. Newer ML/DL network approaches were investigated from a security standpoint [35]. Massive data-imposed limits on Intrusion Detection, necessitating the choice of features with good classification efficiency, hence reducing computational costs. One of the statistical ML algorithms in [36], generated a library of Hidden Markov Models (HMM) patterns and 2 architectures with different performances that can detect and

track the course of attacks in real time. Feature grouping based on the linear correlation coefficient (LCC) and cuttlefish algorithm (CA) [37] have been applied by the DT classifier.

The dimensionality reduction methodology was combined with the (information gain (IG)) method and (principal component analysis (PCA)) for IDS system utilizing a hybrid approach that merged the two algorithms.

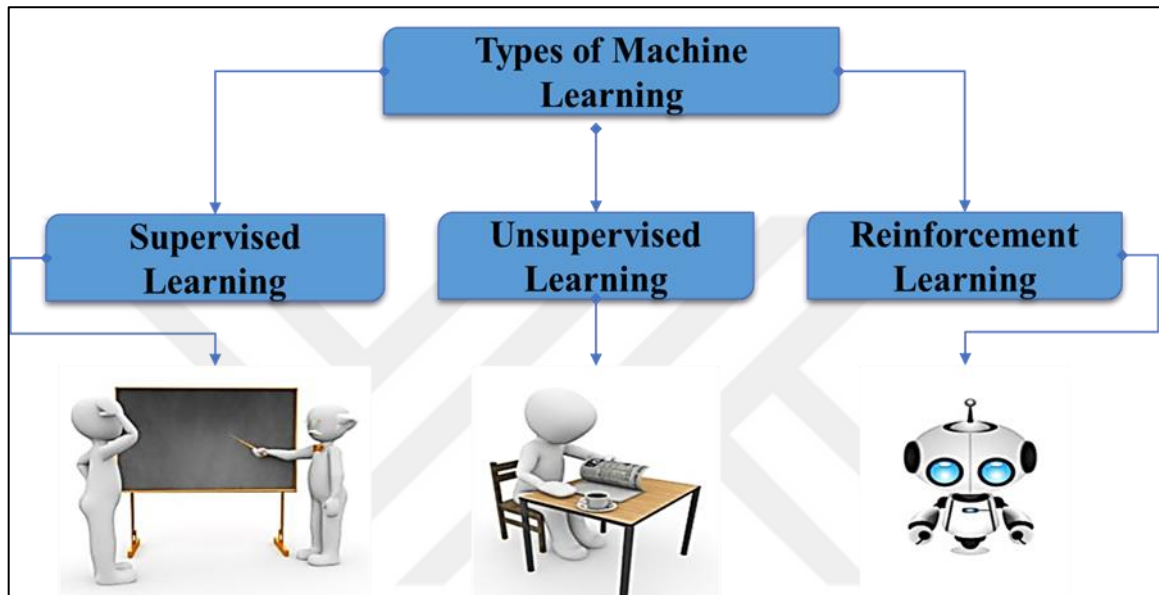


Figure 2.3: Main part of the Machine Learning.

2.2.2 Deep Learning Technique

Deep learning is a subset of ANN, a type of ML capable of multi-level information comprehension [38]. Identification and segmentation have been utilized in numerous fields of study, including transport and routing systems and healthcare. Data for object recognition, language translation, speech recognition, and decision-making are all processed in a manner reminiscent of the brain's information-processing and communication apparatus. In WSNs, DL manages defects, anomaly recognition, energy gathering, data effectiveness computational, as well as routing. The utilization of deep learning models has improved the data security, categorization, and prediction operations of numerous detection systems, malware scanners, and spam filters. Various intelligence applications give a structure for distinguishing between "normal" and "malicious" examples, like attacks and regular packets. As the number of available deep learning models grows exponentially, so do the capabilities of attack planning tools.

Additionally, it's a crucial tool for boosting IDS performance, as it gives associated IDS definitions, clarification of IDS kinds, and its applications [39]. Using the benchmark NIDS dataset NSL-KDD, NIDS has developed sparse auto-encoder and soft-max regression to verify anomaly detection accuracy. Software-Defined Networks (SDNs) exhibited the potential to establish a robustly secured net, reducing attack possibilities dangerously, with the explanation of the potential use of DL for flow-based anomaly detection systems [40]. Also, emerging security vulnerabilities inside the IoT architecture's layers have been uncovered by a survey [41]. Dealing with enormous data is difficult for IDS; however, machine learning's capacity to deal with massive data aids in overcoming this obstacle. Unlike machine learning, DL can extract features automatically without requiring feature engineering [42]. Therefore, the feasibility of inefficient deep learning routing has been demonstrated and explored in recent DL work using network anomaly detection [43]. Some IDSs implementing deep learning techniques for intrusion detection demonstrated limits, benefits, and drawbacks [44]-[47].

The recommended CNN-based intrusion detection method automatically performs complex features in constantly changing environments, which is deemed essential for network IDS. DNN-IDS boosted user confidence and communication since the black-box nature of DNNs limits DNN visibility, and IDS is vital for building trust. By training DNN for Intrusion Detection Systems (DNN-IDS), the illustrated input features are optimal for identifying any type of intrusion [48].

Hierarchical spatial-temporal features-based intrusion detection system (HAST-IDS) was used to develop a unique intrusion detection system. High-level temporal characteristics were then trained using LSTM [49]; The precision and accuracy of three models, a (vanilla deep neural net (DNN)), a (Self-Taught Learning (STL)) technique, and a (RNN-based LSTM), are evaluated. Utilizing (Bi-directional LSTM-RNN (BLSTM-RNN)), an unique deep learning system proposed a concept for the young network to discover attacks using (Bi-directional LSTM-RNN) [50].

By introducing a novel system that includes the design of 'Particle Swarm Optimization (PSO) with adjustable inertia weight and a learning component, the network topology of DBN is optimized. The PSO is then constructed based on the behavior of fish swarms [51].

Scale Hybrid IDS Alert Net (SHIA) is a system for real-time monitoring network and host-level events using hybrid deep neural networks (DNN). This SHIA is watchful for prospective cyberattacks [52]. Several autoencoders are embedded with convolutional RNNs in the AE-based DNN architecture [53] that elicits information of the anticipated connections between the primary features (spatial features) and their temporal evolution (temporal characteristics). Figure 2.4 illustrates a categorization of DL used to detect assaults.

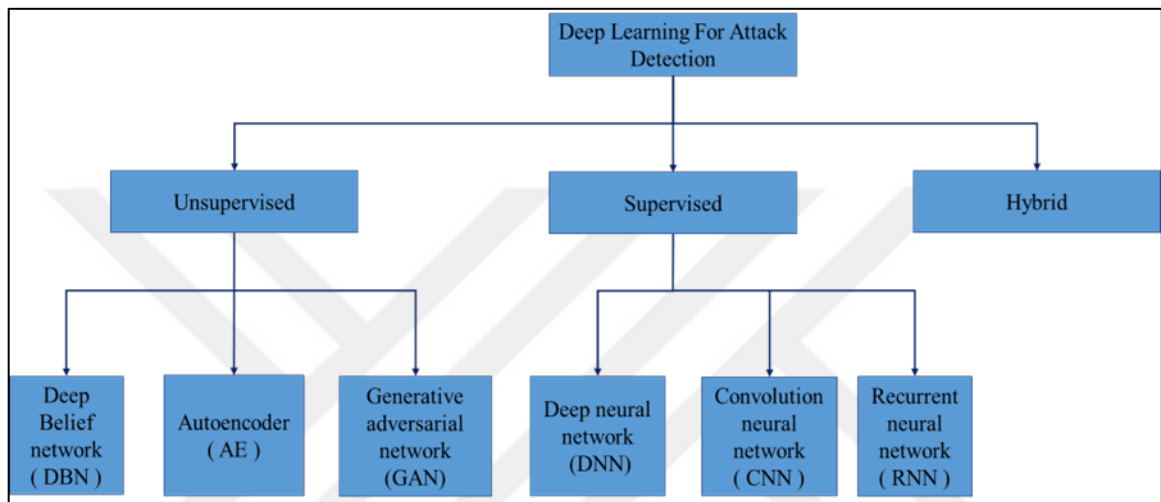


Figure 2.4: General Categorization of DL [53].

2.2.3 MANET with IDS that are Deployed by ML and DL Techniques

IDS provides a supplementary line of defense and serves as the basis for incident response [54] in the event that preventative measures such as authentication and access control fail. After establishing a foothold in a network, an attacker has the opportunity to perform surveillance and identify valuable assets. Even if the initial infection goes undetected, it is possible to prevent further damage by identifying the attacker's lateral movement. IDS plays a critical role in recognizing such conduct and greatly aids in reconstructing it for future forensic investigation.

Only one-third of organizations independently uncovered the infiltration. According to their most recent data from 2019, the median discovery time in 2016 was 99 days; in 2017, it was 101 days [55]. Even while the reduction in detection rate is a significant step, It remains extremely lengthy. One hundred days is a large period of time for an adversary to uncover important assets and take action [56]. Nevertheless, privilege-abusing insiders can

compromise even the most modern and secure networks. Analyzing vulnerability visualizations reveals a worrying trend: the annual total number of vulnerabilities is rising, and the vast majority are graded as medium or high severity. Figure 2.5 demonstrates the architecture for providing wireless MANET networks.

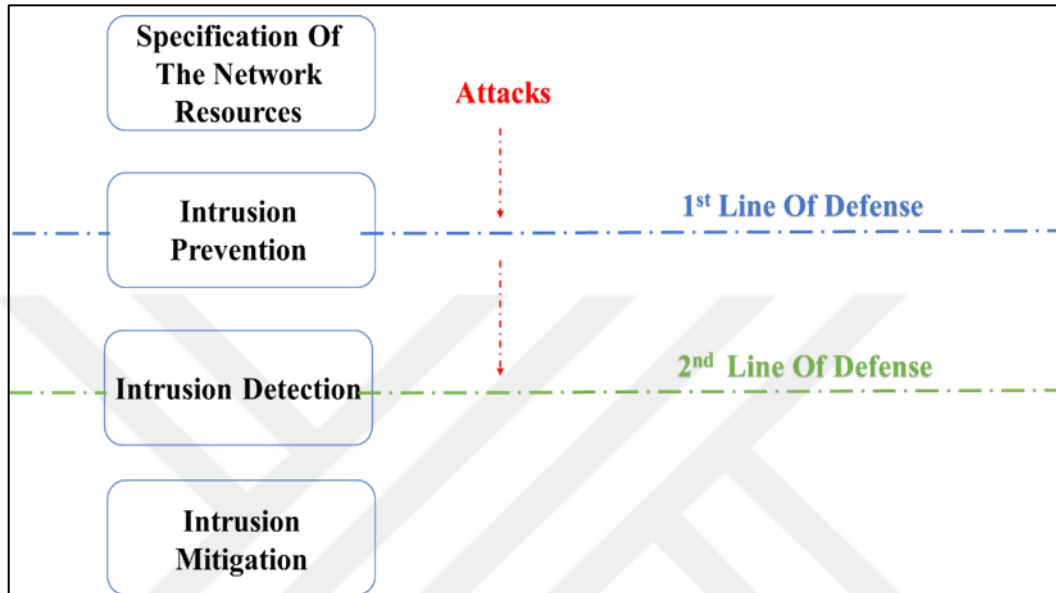


Figure 2.5: Schema for Setting up Wireless Manet Networks [56].

This investigation aims to develop and test an up-to-date, methodical strategy for gaining access to network protocol-specific information in the context of anomaly-based IDS algorithm development in wireless MANET networks [57]. In this context, "modern" entails the following features of the existing framework: Implementing a wireless MANET network requires careful consideration when deciding on a programming language [58]. This language should be compatible with the most popular platforms, provide memory safety, and use a concurrent programming paradigm. When exporting data, it's important to choose a format that keeps the data's safety, reliability, and structure intact while being platform-agnostic and able to handle host traffic to feature [59].

The wireless Mobile ad - hoc network implementation should be up to date so that it can use multi-core computers and scale to accommodate massive workloads. In addition, everything needs to be meticulously documented for future academic use. In a wireless network, the

implemented technique needs to work with dump files and real-time data from a network connection, and it also needs to have a command line and a library interface.

In addition, this strategy will focus on data accessibility and adaptability in order to improve the workflow and efficiency of wireless network experiments, from host traffic to characteristics [60], in contrast to present systems that save data minimization. This is consistent with the notion that a net feature-collecting system would have to offer information at the system's most fundamental level before proceeding to higher-level abstractions. Figure 2.6 depicts information standards IDS in wireless Mobile ad hoc networks. This strategy should be tested using current data and a categorization scheme based on wireless network anomalies.

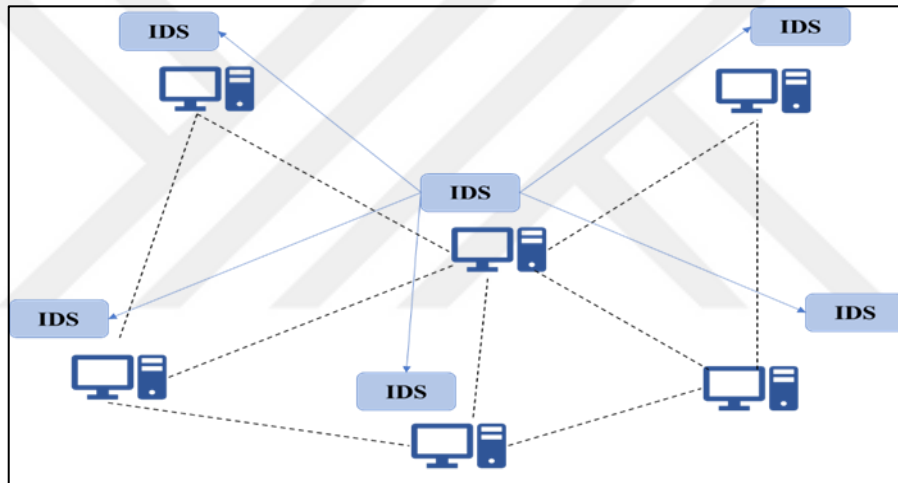


Figure 2.6: IDS for Information Delivery in Manets.

MANET is typically used to analyze one or more physical characteristics over a vast area. A MANET is composed of numerous tiny nodes (sensors). Such devices detect the region's physical characteristics, including temperature, pressure, humidity, and air pollution. They are applied in numerous applications, including the monitoring of natural disasters such as floods, earthquake, and other geological calamities [61], also are utilized for the army, robotic, automated warehousing, industrial, and medical applications. MANETs are commonly used to measure temperature, altitude, humidity, and other natural catastrophes.

Assume each sensor is a simple device with a restricted energy and memory, a method for transmitting and receiving data with the other nodes, and a component for sensing the

characteristics of the outside environment. For the system to work in 95% of Mobile ad-hoc applications, all sensor positions must be known [62]. Without knowledge of the actual location of the sensors, their data is reliable and of limited utility. In numerous applications, the exact position of only just few nodes may well be known beforehand.

Consequently, the remainder topology can be completely arbitrary and ad hoc. Many studies investigate MANET-related issues. MANET is an ongoing research field. The majority of these experiments have addressed the issue of delivering per-node location data; Figure 2.7 depicts the structure and flow of collecting IDS features in a MANET network.

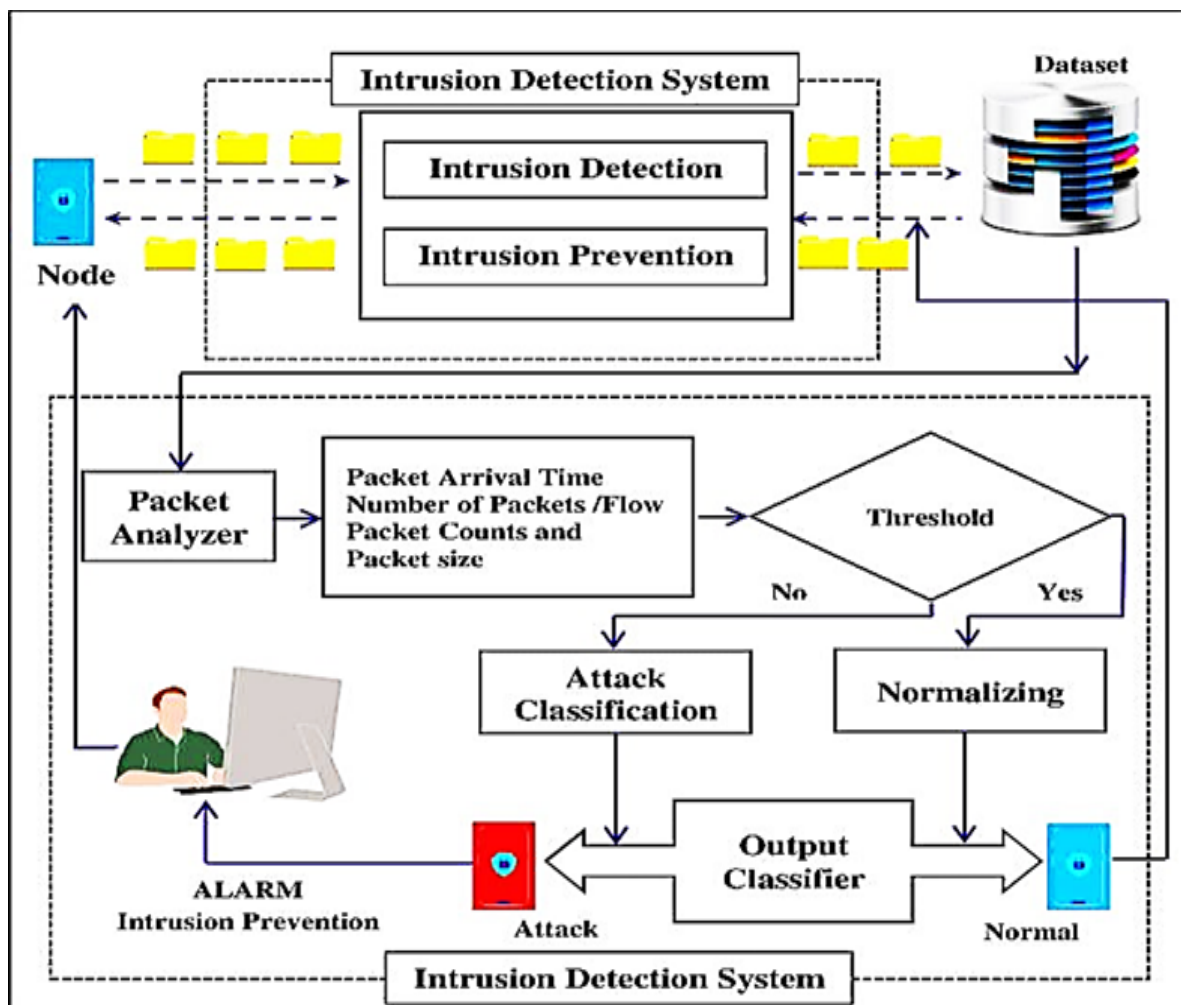


Figure 2.7: IDS Network Architecture and Data Flow in Manets [62].

Multiple real-time attacks in wireless networks are being investigated by looking at data from the network and the application layers. Moreover, in the research [63] only a 2-node clustering approach was utilized to compare to earlier efforts on application layer data.

Information from the application layer was also a part of the probe. Synthetic data was used and retrieved from an open-source repository. Our data comes from a non-synthetic source, a publicly available repository for wireless (P2P) systems [64].

Analyzing the time needed via each algorithm to perform the event is essential to our research, as we saw that the MANET-ANN procedure requires the minimum execution time. As noted in the literature study, the challenges in wireless networks lie in discovering calculations that can manage massive quantities [65].

In mobile ad-hoc, the notion of avoidance is insufficient from a safety standpoint; consequently, the intrusion detection system is an additional concept for facilitating network security; this intrusion detection system helps spy on these malicious nodes within the network. IDS are a one-of-a-kind method used only to identify harmful activity in mobile ad hoc. An intrusion detection system must be constructed as any other software used to forecast malicious network nodes. Because of the centralized nature of the MANET, IDS are being placed on these nodes to circumvent these issues [66] , [67]. The primary MANETS categories are signature-based, specification, and anomaly detection.

Similarly, the intrusion detection system has attack types from the perspective of MANETS, as described in [68] , [69]. The writer has analyzed IDS strategies established on intrusion datasets; utilized the Wi-Fi intrusion detection dataset. From this dataset, feature-based reduction approaches are evaluated using machine learning algorithms. With feature reduction, chi-squared statistics are used to evaluate the accuracy of a dataset [70]. The feature reduction technique yielded a significant improvement. Results are displayed via feature reduction; the 2.4% decrease from 110 to 41 could result in less reliable findings [71]. Misuse-based, anomaly-based, and hybrid intrusion detection techniques have been developed.

These IDS techniques had been misused and now needed to be verified by network packet flow while steered by a set of known, diverse threat patterns. These methods are more precise but easy to apply and don't catch sneak attacks [72]. With a lesser level of precision, anomaly-based detection can nevertheless discover previously undisclosed dangers. The effort is continuously put into analyzing data sets for ways to improve efficiency. It is proposed [73] to use an evolutionary algorithm to choose the best protocol in each network environment. DSR, OLSR, and AODV are all employed to their utmost potential.

The efficiency of each routing protocol in the WPAN is optimized using fuzzy logic techniques and fuzzy neural approaches. Similarly, a genetic algorithm improves efficiency in yet another investigation.

Finally, this approach has increased the accuracy and performance of a support vector machine. MANETs in an open medium and a changing topology with insufficient resources and inadequate current supply are subject to assaults based on their fundamental properties. This paper explores how a safe IDS based on the dynamic resource routing protocol can detect compromised nodes in MANETs (DSRP). As a result, DSRP can permanently sidestep the issue of false acknowledgement [74]. The DSR protocol, widely used in wireless and wired networks, is considered one of the proposed approaches for dynamic source routing. The two main parts of DSRP are described below.

The system incorporates a multilayered defence system. The intrusion detection analyzes the audit data and determines if a server is about to be attacked based on the data's explanation. It matters if IDS is expected to be network-based or host-based. Network-based IDS keep a watchful check on the network gateway thru which packets are routed into the network hardware. Host-based relies on the operating system to review and analyze the data [75]. In Table 2.1, we see various assaults that can be conducted against a MANET and the importance of the different security precautions that can be implemented in response. They utilized to evaluate the performance of the suggested Machine learning ensemble approach; 'V' indicates that it is vital, and 'Not' indicates that it is not.

Table 2.1: Effect of Different Types of Attacks in Manet

Types of attack	Describe within the Network
DoS	Not
PROB	Not
U2R	Vital
R2L	Vital

2.3 STUDIES COMPARISON

A comparison has been made between the works focused on the security of MNET and intrusion detection on these networks through machine learning (ML) as well as deep learning (DL) to analyze the network traffic. It was found that the experiments using available or unreal data were less accurate when compared with using actual data to analyze the network traffic and detect intrusion. Work in this study used a real data set of the MANET network. Table 2.2 demonstrates the type of intrusion detection and common attacks of Manet networks using machine learning algorithms. Table 2.3 lists the accuracies of the attack's detection to the available data that have been used, the algorithms used, and the number of hidden layers.

Table 2.2: Types of Intrusions and Common Attacks.

Researchers	Type of attack	Type of IDS
Research [39]	DoS	CIDS
Research [41]	Remote-to-User-Attacks	HIDS
Research [42]	DoS	G. IDS
Research [72]	Normal net and Attack net	G. IDS
Research [56]	cyber-attack	G. IDS
Research [40]	Web Attack	S-A IDS

Table 2.3: Comparison With Previous Works.

Researchers	Type of data	ML algorithm	No. of hidden layer (DL)	Selected Features	Test Options	ACC %	Year
Research [42]	CICIDS2017	Decision tree	2	dimensional features & Five categories classification.	Random	79%	2020
Research [72]	NSL-KDD dataset	Random tree + NB Tree	/	Correctly Classified (CC)	10-fold cross-validation	80%	2017
Research [59]	ICS	SVM	/	Correctly Classified (CC)	10-fold cross-validation	79%	2018
Research [73]	Waikato	RF	/	Correctly Classified (CC)	10-fold cross-validation	79%	2020
Research [39]	NSL-KDD dataset	CNN	2	No	10-fold cross-validation	77%	2019
Research [41]	NSL-KDD	Bagging (REP-Tree) Classifier	/	No	3-fold cross-validation	81%	2015
Research [40]	CIC-IDS2017	DBSCAN Classifier	2	Bwd.Packet.Length.Min and other	10-fold cross-validation	82%	2021

3. RESEARCH METHODOLOGY

3.1 INTRODUCTION

Considering the importance of MANETs in all facets of daily life. This made it vulnerable to many numbers of attacks because of the absence of a central authority between nodes of the network, the use of ready data, the imprecision of their dependability, and the difficulty of analyzing it. All these issues lead to against the network many attacks. Therefore, we proposed creating a network with applying an intrusion detection system (IDS) on it and collecting the output data as a dataset by using NS2 instead of ready relying on data. Then, training this data by using deep learning (DL) algorithms. Thus, in this work, a current investigation aims to determine APR and E2E delay of nodes in MANETs as well as detect attacks that cause nodes to drop due to attacks. In this experiment, ANN are used to improve the detection of drop down packets during sending. Since, ANN is overmuch accurate at detecting malicious MANET activity because each communication is recorded using a memory frame.

So, this chapter will be divided into two parts; The first part will explain the effect of IDS on detecting attacks on data through the NS2 program, where the MANET is configured to be vulnerable to a kind of DoS attack, specifically a (black-hole) attack. And detection system is included on a network to study attacks on the transfer rate of the data and the delay time for use in the next part. The second part will present ML and DL, choosing the appropriate algorithms for the data of the MANET network. Three DL algorithms were selected to measure the accurateness of each one and a speed of attack detection. The protocol for this network (AODV) With an intrusion detection system was trained through these algorithms. Each part of the working methodology will be described in the following chapter.

3.2 THE FIRST PART, CREATING THE DATA OF THE MANET NETWORK

In the framework of wireless ad hoc networks, a sensing medium must be covered with several nodes. Once a sensor has data, it can send it to a server or sink node. The server node is a powerful node that can handle multiple network data flows. Computer virtual environments simulate wireless network scenarios with a much lower budget than real-world

experiments. Using the simulation (NS2.4), different types of sensors can be created with adjustable parameters for optimal sensing. MANNASIM, an extension of the network simulator (NS2.4), can simulate wireless sensor networks for this project. MANNASIM's classes can be changed to simulation wireless sensors (MANNASIM). Network monitoring, which logs packets and other analytical details, enables adjustments to back-office processes in protocols and standards [76].

MANNASIM is a wireless sensor network simulation environment. This tool's framework simulates wireless network scenarios using NS2.4. Script generator tool (SGT) works with network simulator (NS2.4) framework codes to create simulation results and create simulations. Thus, a TCL file is the starting point for any simulation. Since the underlying framework is available under an open-source license, users can modify it to fit their needs better, which contains the general wireless network scenario. Script generator tool (SGT) monitors TCL procedures soon after. MANNASIM is a JAVA platform integrated with network simulator NS2.4 that simulates wireless sensor networks with different sensing requirements.

3.3 ROUTING PROTOCOLS TYPE REACTIVE (AODV)

In this simulation, a routing protocol was used AODV, it's a protocol determines a path through which data packets travel from one computer to another. This protocol generates routes based on when a node detects it requires an administrative district. By transmitting Route request packets, it quickly identifies a path to follow.

3.3.1 Attacking Model (Blackhole attack)

A blackhole attack is used to model simulation, one of the attacks common that MANET networks are exposed network. In this assault, The attack node utilizes a routing protocol route (AODV) discovery method to promote itself as the closest path to the destination node (DN) that wants to receive its packets. It broadcasts to other nodes on the network its availability of new routes, regardless of whether it has checked its routing database; this ensures that the attacker's nodes are always available to respond to route requests and intercept and store data packets. The blackhole attack falls into two distinct kinds [87]-[89]:

- a. In the first kind, it is simple for a single black hole attack to occur in MANETs. In a single blackhole attack, just one rogue node operates within the network.
- b. In the second kind, Multiple blackhole attack nodes inside a network function as malicious nodes in a coordinated blackhole attack. If a network node wishes to transmit packets, it must first send the route demand message to every one of its neighbors, including an attacker node. The attacker node reacts speedily to the source node (SN) via transmitting the route response message. When it gets RREP, it transfers packets along the path of the attacking node. The blackhole node begins discarding all incoming packets after receiving them. We have included this type of attack (blackhole) in our network for procedures testing on it and checking network performance with the existence of an intrusion detection system, as will be explained.
- c. MANET malicious nodes will attack via direct link (single hop) or re-transmission (cooperative transmission through multiple hops). MANET's blocking malicious nodes issue has changed. Thus, recognizing the attacker node is less likely when both the victim and attacker nodes move. The attacker and victim nodes may not always be directly connected. When a direct link fails, an attacker node attacks the relay node indirectly (see Figure 3.1).

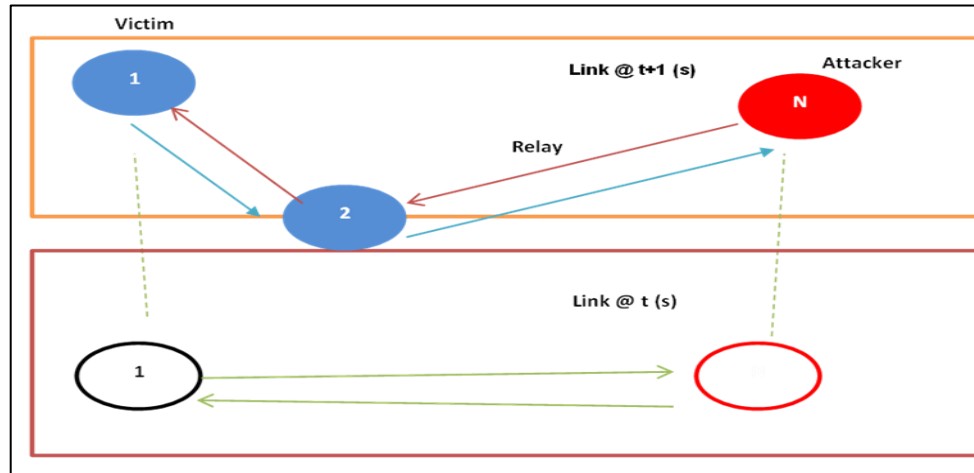


Figure 3.1: Demonstration of Victim-Attack Mobility at Two Different Times of Link Duration.

3.3.2 IDS-AODV Model

AODV routing protocol with IDS system has been used to monitor attacks (blackhole attacks) in this step. IDS is the system used to generate reports after evaluating malicious and normal nodes' network behaviour. Where assessing intrusions affects systems' availability, confidentiality, and integrity. IDS nodes initially execute the monitor mechanism function in sniffing mode. The IDS can determine the value of a suspect node depended on the difference in routing info transmitted from the suspicious nodes. A node can be considered a black hole if the routing information exceeds a predetermined threshold. Then, a normal node inserts the attacking node, which is recorded in a block of a message broadcast by the IDS node, to its block table, and it begins routing by forwarding an RREP packet. If the normal node receives an RREP message from a node listed in the block table, it will discard this RREP message to prevent malicious traffic from affecting the network.

Figure 3.2 depicts the function of a blackhole attacker node, which is dependent on AODV routing. Node (S) represents the sink, which is a source node (SN) that wishes to deliver packets to a destination node (DN) and (nodes X, Y, and Z). Assuming node X has a route to node S, it will initially send an RREP message to node S. Before this, the malware node M sends a bogus RREP message to node X, informing it that its serial number is better than that of a diverted from node S. Assumes node X to be the shortest path to a destination node and transmits packets through it. The attacker node M will then receive the packets and drop them.

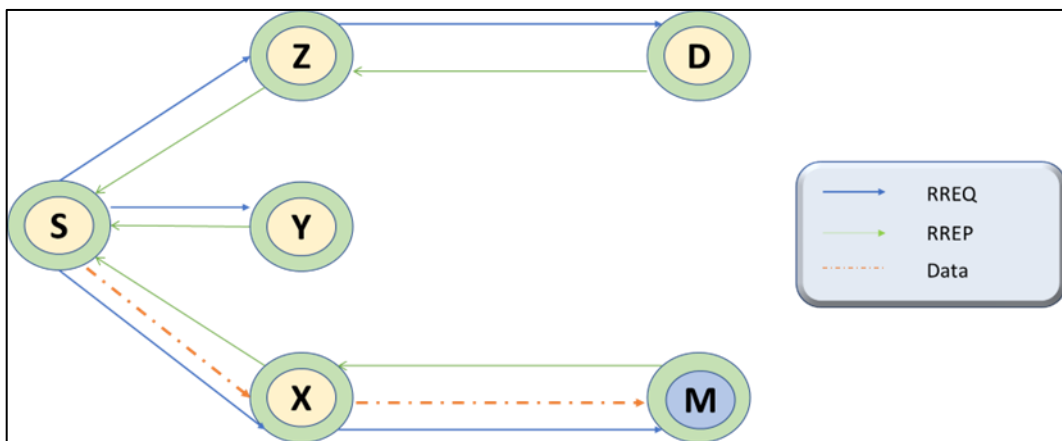


Figure 3.2: Blackhole Attacker Node Function in the Ad-Hoc Network.

3.3.3 Details of ID-AODV and Network Monitor

The specifics of the planned IDS-AODV's design and execution are now presented. Because of a distributed nature of the MANET networks, a centralized IDS node can't monitor the entirety of a conversation. As a result, distributed network monitors must track the RREQ and RREP messages in a demand sequence (NM). The structure of a network observer is shown in Figure 3.3. IDS-AODV routing messages are passively listened to by network monitors to identify faulty RREQ and RREP messages. Messages are organized into streams according to the request-response pattern they were sent. The RREQ ID, along with the IP addresses of the sending and receiving nodes, can be used to trace the origin and destination of any request/response pair.

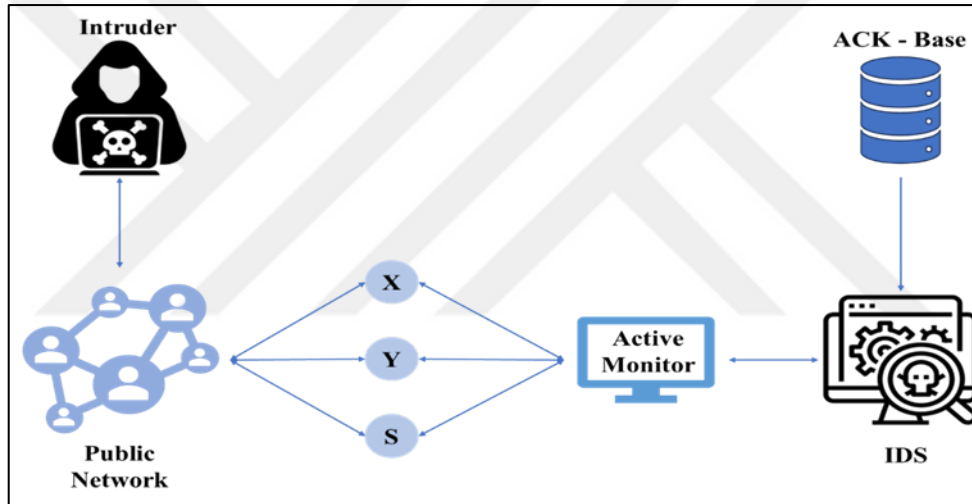


Figure 3.3: Structure of Ids-Aodv.

3.3.4 Procedure of the Proposed Method (Creating Dataset)

- a. The simulation is created via the 50-node communication model. For wireless sensor network specifications, a wireless channel with bidirectional signal transmission is established. IEEE 802.11 MANET networks employ a message architecture devoid of infrastructure.

- b. Been use a black hole attack; it simulates four possible malicious actions through four malicious nodes; Then, the scenarios calculate the network ARP and E2E delay.
- c. Location of attacks nodes is determined at random so that we can acquire rare outcomes at all-time and run a simulation; nonetheless, the attack's impact varies depending on the location of malicious nodes.
- d. Now been implementation of a mechanism to identify these malicious nodes is required. After analyzing the findings of the trace, we employed a detection method that leverages these 2 trace files to identify malicious nodes. The system processes the information gathered from the two trace files. A first trace file is employed to define behavior for normal scenarios.
- e. In a harmful situation, select nodes are set as harmful; in this trace file, the behavior of those attack nodes is compared to legal behavior, and it is found that such a trace file includes blackhole attack behavior patterns for malicious nodes.
- f. This pattern of conduct is recognized as a signature of assault. This assault signature is used to identify malicious nodes. When a user provides a scenario, its trace file matches a previously constructed attack signature. If this signature matches the trace findings of any nodes in each scenario, it asserts that those nodes are malicious.
- g. The basic approach to determine if a node is harmful is to compare an attack behaviour with a normal scenario. A node misbehaves when it drops packets based on its behaviour. While other nodes correctly behave. In the end, the data is collected from this scenario to compare its results after training the deep learning algorithms. Figure 3.4 explains flowchart IDS routing on (AODV).

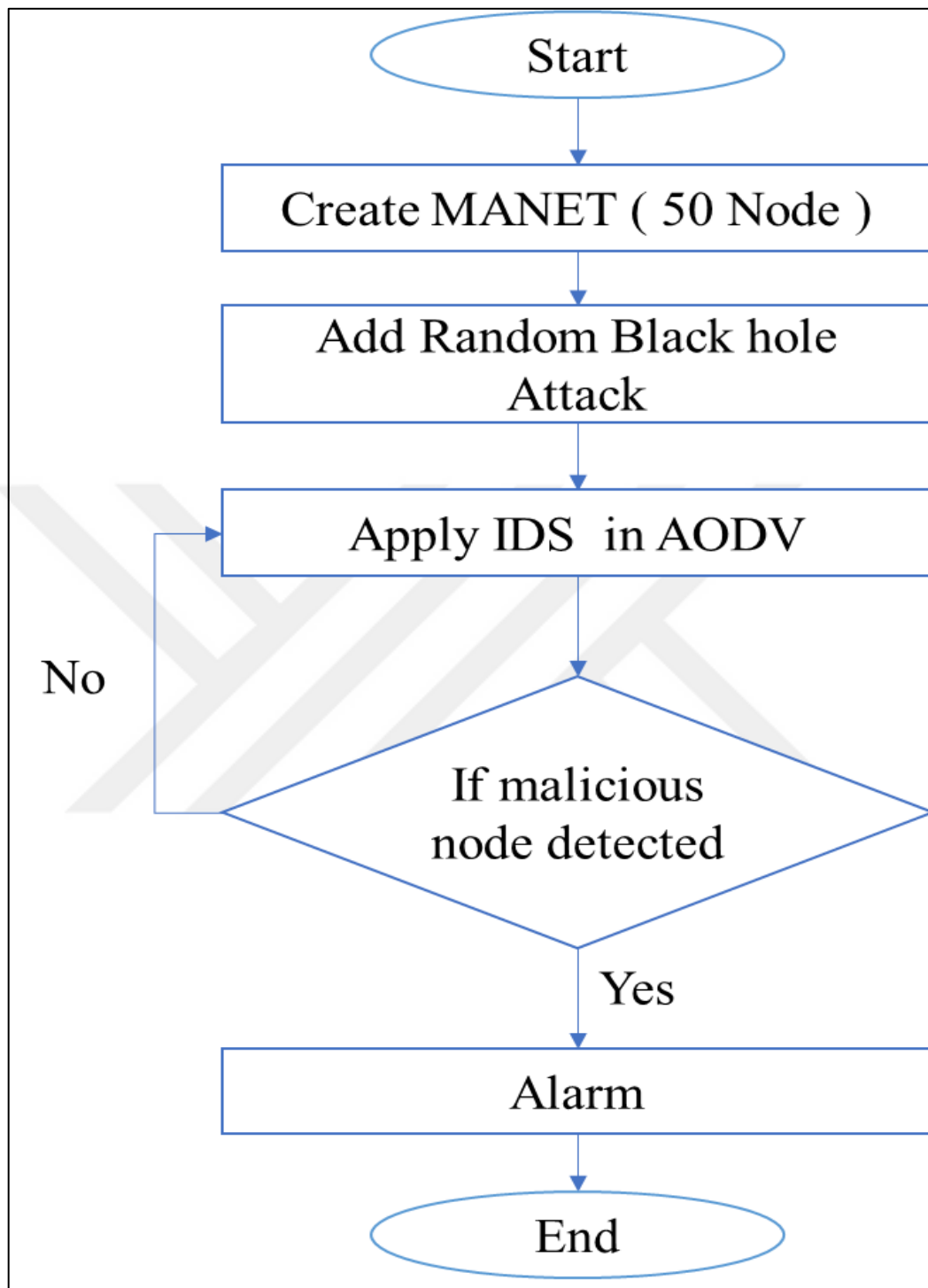


Figure 3.4: Flowchart to the IDS with the Routing Protocol (Aodv).

3.4 THE SECOND PART, THE PROPOSED METHOD FOR TRAINING DATA BY USING DEEP LEARNING

The suggested model was implemented utilizing DL depended on a distributed intrusion detection system (IDS). It is possible to set transmitting limits regarding dropped and delayed packets in MANETs, where DL is used to train the IDS system to improve accuracy and performance. This is because deep learning can analyze data, detect attacks, and solve many issues. This will produce false alerts in the MANET context due to the wrong recognition of normal as virulent node patterns. Therefore, CBPNN, FFNN and CNN are implemented in machine learning systems, respectively, to avoid the problem and manage the dynamic network architecture and mobility of nodes. The following elucidates the operational procedure of the model that has been proposed by using deep learning (DL):

- a. After MANETs are configured with prying nodes, an attacker and a packet dropper line generate erroneous data packets. The data transference is started by the source node, which also begins the procedure for pathfinding.
- b. All nodes transmit data through the request forwarding (AODV) routing protocol under the monitoring system (IDS). Targeted destination nodes generate route replies; packets are forwarded via intermediate nodes till they reach a source node. When a route is discovered, source and intermediate nodes are responsible for data packet forwarding.
- c. By monitor of the nodes by the IDS system to keep tracking routes and responses and performing physical layer eavesdropping to observe a reception is redirected forwarding counts packets of data, and the wrong packets created via the nodes are tested by calculating the E2E and ARP.
- d. Initiating the training routing protocol (AODV) with IDS to detection procedure with CBPNN, FFNN and CNN algorithms based on deep learning (DL), where the DL algorithms learn from data to detect attacks. The attack detection accuracy fluctuates due to attacker node behaviour uncertainty.
- e. Supervised learning is started by providing both the input (r) and output (T) vectors. Changing the weight coefficients between layers reduces error (E2E and APR). The error was found by correlating the resulting and routing. The Random variable at s^0 , see in Eq (3.1 and 3.2).

$$R = \text{net}(r) \quad (3.1)$$

$$R = W \times r + b \quad (3.2)$$

In (Eq. 3.1), R corresponds to the output vector, while b represents the model's bias (Eq 3.3). This means that the coefficients of W may be tweaked by the net in order to maximize the correlation between R and T. The learning process entails locating the lowest possible value of Eq (3.4).

$$E = R - T \quad (3.3)$$

$$MSE = \frac{\sum_{n=1}^i e(n)^2}{i} \quad (3.4)$$

MSE is a training/learning performance statistic; the net is trained to guess the velocity that best recognizes attacker node activity (Eq. 3.5) used to define attack recognition accuracy.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+PN} \times 100\% \quad (3.5)$$

- f. Traffic congestion and malicious behavior can be distinguished using estimated values of E2E, ARP, RMSE rate, MAE rate, and MSE rate for algorithms, as well as the forwarding ratios of data and control messages and the dependability between original and fake packets.
- g. If a virulent node is revealed based on the trust factor and other input parameters, it is decided whether the packet must be delivered, or a partition must be constructed to bypass the connection. Alterations are made to the connection path and drop the nodes that include attack. In this manner, algorithms are implemented in the desired shape by continuously analyzing and predicting nodes depending on decisions. Figure 3.5 shows a flowchart presenting the proposed model.

3.4.1 Algorithm

- i. Generate a MANET of 50 nodes that are configured in a random design and get an input limitations & reward R_t for time t .
- ii. Choose the source (S), destination (D), and mobile node from the list of nodes.
- iii. Apply IDS, the values of the inputs, to find the node whose state s_t
- iv. Select the action with the most significant D ($S_t; a_t$) or a decreased E2E for indiscriminate (random) activity.
- v. Calculate D ($S_{t-1}; a_{t-1}$); D ($S_{t-1}; a_{t-1}$) and then update it if the correct course of action is selected.
- vi. Even when a node detects packet loss, the data still needs to be rerouted using pattern recognition neural networks so that a different node in the network may receive it.
- vii. And then, from the source (S) node, restart the transmit.
- viii. To the record list for the sub-node that corresponds to S_{t-1} , append D ($S_{t-1}; a_{t-1}$)
- ix. Select if S_{t-1} must be distributed into two via investigation of a track record list for S_{t-1} .
- x. Partition: =T % if record list length is less than record list mini size.
- xi. In the recorded set, find the mean and standard deviation of D ($S_{(t-1)}$); ($a_{(t-1)}$)
- xii. If the average is less than that required, then divide into two groups: =t and =T
- xiii. Update the values of R_t , a_t and S_t For use in the next iteration or perform partition if necessary.
- xiv. Back to a_t .

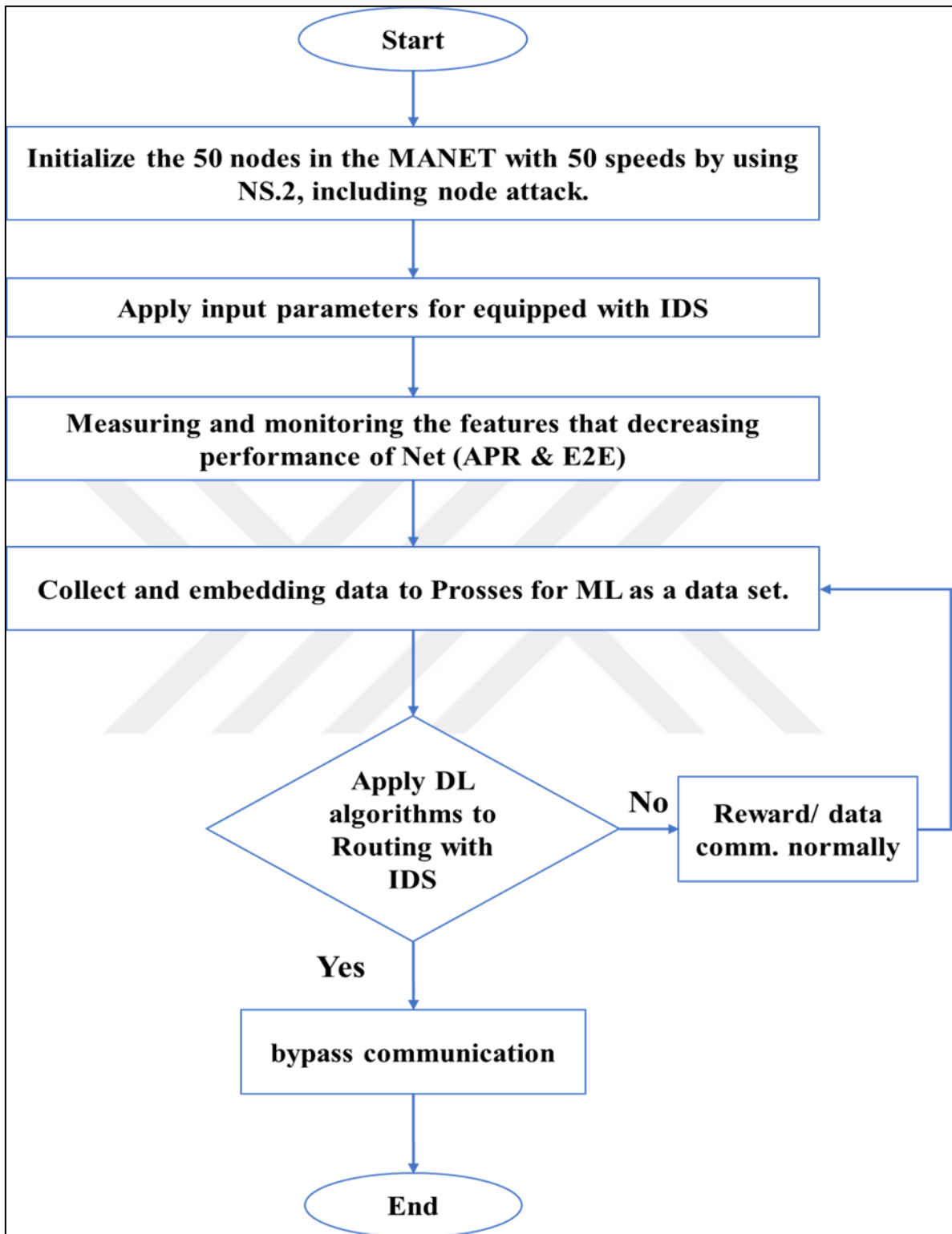


Figure 3.5: Flowchart Describe the Proposed Model.

3.4.2 Implementation of the Algorithms

Based on the studies that rely on deep learning, which were described in the preceding section (chapter 2), the DL process based on ANN is as follows. It's given that the human brain is made up of neurons, a single neuron gets 1000 of signaling from different neurons; it is constituted of node layers. These nodes connected within each layer are related to the side by side layers. The network's depth is proportional to the number of its layers. In these networks, signals are transmitted among nodes as well as allotted weights. The node within a mass will exert an important impact on a subsequent nodes layer. A last layer collects weighted inputs to create the end product based on the weight (cost) function curve, as demonstrate by Figure 3.6.

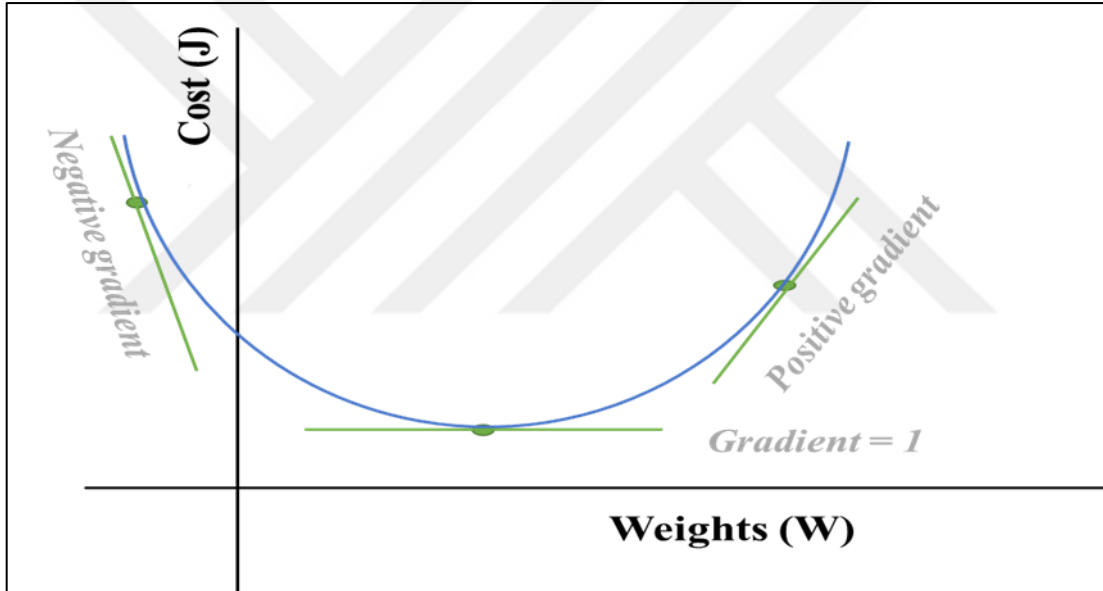


Figure 3.6: Weight Function Curve.

Deep learning (DL) necessitates vast amounts of data to deliver accurate conclusions; information is fed as massive data sets. Neural networks may classify data during data processing based on the responses to a sequence of binary valid or invalid questions, which requires incredibly complicated mathematical calculations. Figure 3.7 depicts the mechanism of the work of deep learning and based on the mechanism of deep learning (DL), the weight of three algorithms (CPBNN, FFNN and CNN) by using (Eq 3.6).

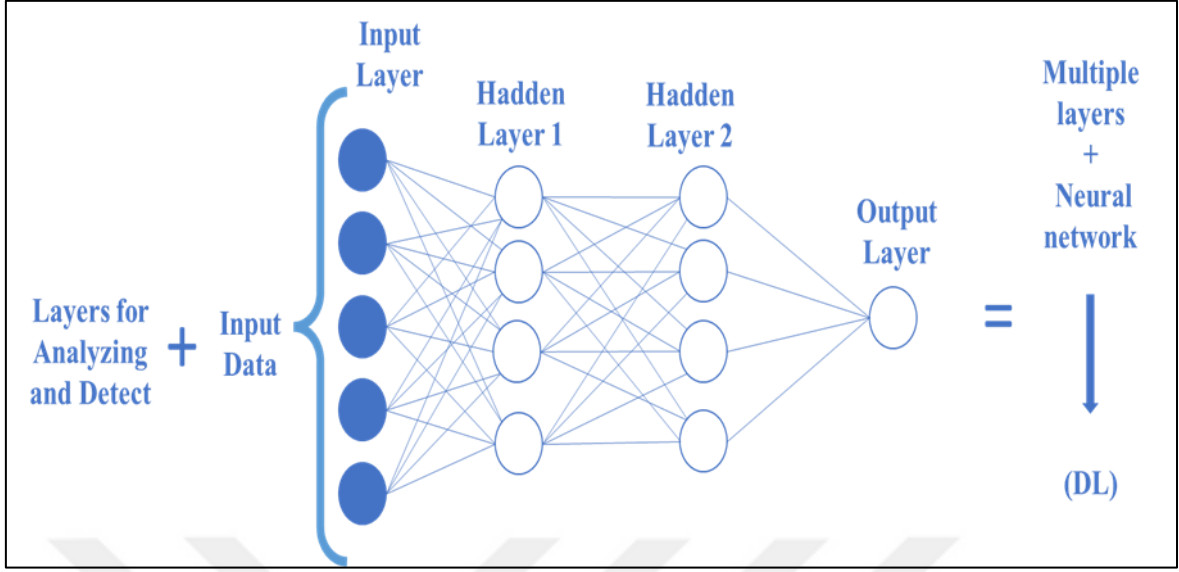


Figure 3.7: Display of the Deep Learning (DL) Process.

$$W_x \{\text{next weight}\} = W_x \{\text{old weight}\} - A \{\text{DL rate}\} \left(\frac{\partial \text{Error}}{\partial W_x} \right) \quad (3.6)$$

a. First Model CPBNN algorithm

The cascade back-propagation neural network (CBPNN) algorithm serves as the conceptual foundation of a model for hasten accelerating learning in neural networks. The NN will be constructed. After each iteration, the weights of a neural network are altered minutely, built on error rate loss recorded for the duration of the prior iteration. The accuracy is enhanced by refining the model's weights to improve its generalization. On the MANET, neural networks (NN) are trained by presenting the network with training inputs whose exact outputs are recognized. The model (CBPNN) attempts to modify the weights of synaptic contacts in the MANET so that the real yields are closer to the desired yields.

Two of the components, referred to as the "training" and "cross-validation" sets, must have known input/output pairs. In order to prevent overlearning, the cross-validation set permits verification of the input-to-output transformation learned by the network during training. This set may include (either the training set or the cross-validation set); the test set consists of inputs that need to be transformed into outputs. During training iterations, these weights

would be continuously modulated in accordance with the CBPNN algorithm, whose name derives from the fact that it combines characteristics of the back-propagation (BP) and cascade-correlation (CC) algorithms.

As depicted in Figure 3.8, the CBPNN method begins by employing a pseudo-inverse method to calculate the weights between layers of input and output of neurons (where each letter represents a distinct weight). Existing neurons participate in gradient descent learning as long as the training error remains over a defined threshold; consequently, the learning rate remains above the threshold. When the learning rate drops below this limit, a new cell is created. After the quadratic error measurement has dropped to a specified value, the rate of learning is paused. The network then continues to grow with the existing neurons indefinitely. The CBPNN functionality provides two important advantages:

- i. Permits the MANET to get away position minimums of the MSE.
- ii. Speeds convergence by preventing gradient descent in multiple subspaces (useless expenditure of time) equal of synaptic connection weight. The cascade approach focuses learning on the hypercube.

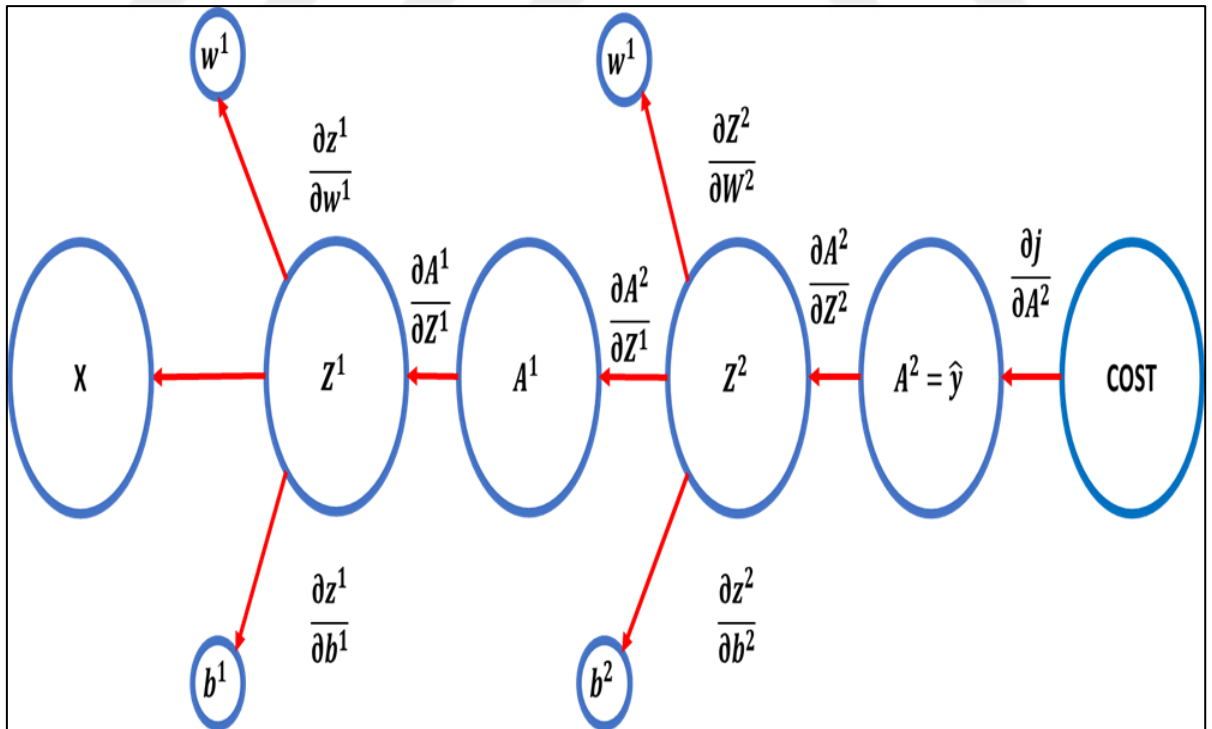


Figure 3.8: Computational of Wight for Cbpnn Algorithm.

The analogue approach provided would give the required high resolution for such computations, as displayed in the above figure.

b. Second Model FFNN algorithm

The FFNN Model is the neural network in which nodes are not connected in a cycle. It's the most fundamental type of (ANN), as unidirectional of the processing of the input data. While data may push through a large number of hidden nodes, it never flows in the reverse way.

In this algorithm, the several inputs are multiplied as they pass through the layer. by the weights; the FFNN method computes the weight as shown in Figure 3.9. The sum of the weighted input data is then determined by adding each individual value. If the values is greater than the predetermined threshold, the output is normally greater than one; if the sum is less than the limit, the output values are generally less than one. In addition, this learning and training procedure results in a sort of gradient descent. In perceptions with several layers, the procedure for updating weights is substantially identical. Nevertheless, the process is more precisely referred to as back propagation (BP) process.

In these case, Each hidden network layer is then updated based on the values generated by the last layer's output.

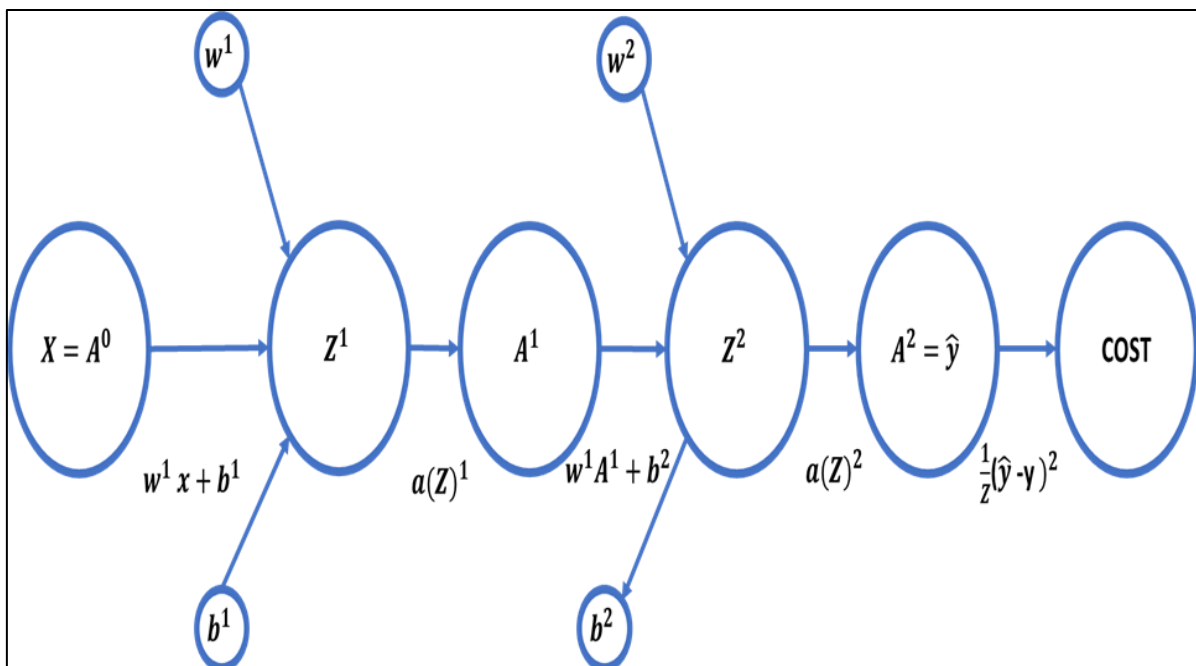


Figure 3.9: Computational of Wight for Ffnn Algorithm.

c. Third Model CNN Algorithm

The CNN algorithm is known (CNN or Conv-Net) is a form of DL with a (FF) design that can generalize well. It is most typically used to assess networking like the network utilized in the proposed study (MANET). This network uses convolution, a mathematical procedure that transforms two functions into a third function that shows how they change each other. This model includes a limited number of computational layers that can learn several abstraction levels for characteristics of diverse input data (e.g., packets). The initiative layers learn as well as extract elevated-level features (at a lower abstraction level) while the deep layers train as well as extract at a low-level features (with an advanced conception). The structure incorporates both convolution and completely interconnected layers. Between these two levels, subsampling layers may exist, likely to experience a wide decentralized input data because of complexity and obtain the best deep neural networks (DNNs) by utilizing new-scaled.

Therefore, convolution neural network is instantly executed in datasets within a moderately significant amount of nodes as well as elements that need to be trained. When input in a ConvNet, each layer generates several activation functions that are passed on to the next layer to calculate the weight, as described in Figure 3.10.

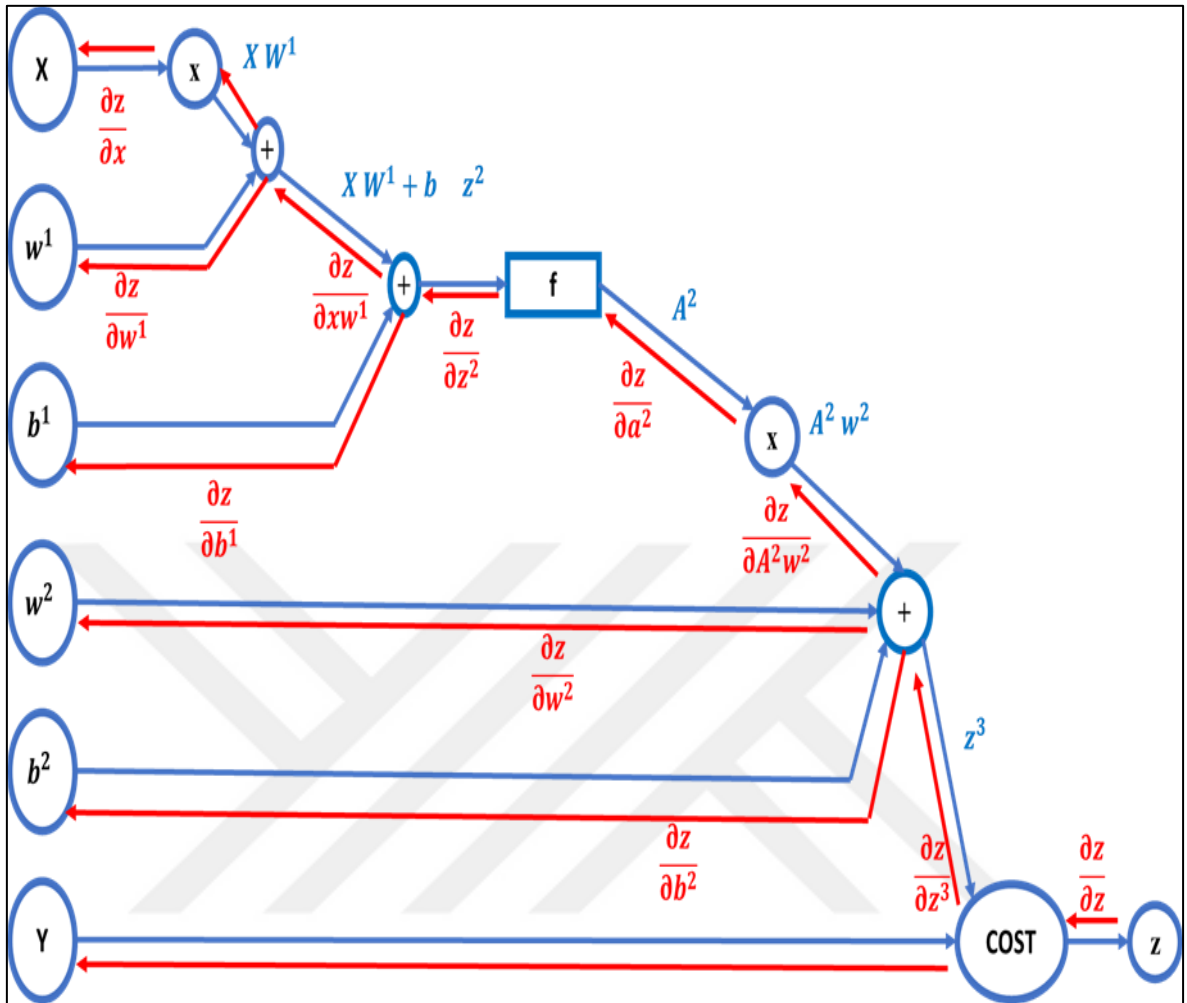


Figure 3.10: Computational of Wight for Cnn Algorithm.

To implement deep learning (DL) in the proposed model, the following steps summarize the procedure of the three algorithms (CBPN, RNN and CNN):

- Net with 1 input layer (500n), 2 hidden layers (10 sets), and 1 output layer.
- In the first layer, all inputs are processed by each prediction before making a decision.
- The next-layer predictions use the outputs from these first-layer predictions to make decisions.
- Similarly, the output layer prediction makes decisions based on the outcomes of predictions in a second hidden layer to arrive at a final outcome.

4. IMPLEMENTATION AND RESULTS

4.1 INTRODUCTION

Similar to the preceding chapter, this chapter will also be divided into two parts. The first section will consist of simulation results for network data by using the network simulator (NS2.4), which was installed on a Linux operating system, and analyzing the attack (blackhole) attack impact on the MANET adding IDS. To evaluate the network performance metric through (E2E delay and APR). In the presence of attacks and the addition of an IDS system.

A second part is the results of the training model and AODV routing protocol for the network in the presence of IDS system. Applying in the MATLAB program installed on Windows to implement three algorithms (CPBNN, FFNN, CNN). Depending on deep learning (DL) and artificial intelligence (AI) to determine a performance of a suggested approach, as well as in order to study an effect on the network and to evaluate accuracy of each model. Moreover, the accuracy of each algorithm will determine the network performance metric by training the network on these algorithms and calculating the APR and E2E. Through the accuracy of detecting attacks and time to detect for each algorithm, we will be able to determine the (E2E delay and APR) of the MANET network.

A device to create data of MANET and implement the simulation with the specifications listed in table 4.1 was used for network simulation, data processing, and training of the three algorithms, each algorithm which required approximately 9 hours. In addition, the time needed for the detection (test) varies between algorithms, as will be explained later in this chapter.

Table 4.1: Specifications of the Device Used to Implement the Training for Each Algorithm.

Name	Specification
Motherboard	ASROCK B550 Extreme 4
Memory (RAM)	32 GB, DDR4
CPU	AMD Ryzen 7 5800X
Technology	7 nm
No. of cores	Eight core
No. of threads	Sixteen threads
Core Speed	4715.4 MHz
GPU	NVIDIA GeForce GTX 1660 SUPER
Technology GPU	12 nm
Memory type	GDDR6 (Micron)
Memory size	6 GB

4.2 THE FIRST PART, CREATING DATA OF NETWORK WITH MONITORING THE INTRUSION DETECTION SYSTEM

The simulation was created for network MANET using the network simulator (NS2.4). The area made for the MANET experiment is (3000 m by 3000 m), with support for four malicious node wireless communications with a bandwidth of 4 Mbps. Node mobility random rate of 60 s, and a transmission range is 500 m. The routing traffic is (AODV) and a packet's size (is 1024 bytes). In this section, there are two possible outcomes, the first of which based on the amount of nodes (20, 30, 50), the second on the amount of node speeds (10, 20, 30, 40, 50). These two scenarios have two objectives.

The first one is to study the impact of the amount of nodes as well as their speed on the network to be monitored using the IDS system in the presence of the assault (black hole). A second one is to process the data of the network that will train in the next part. This is because deep learning (DL) requires more data based on the algorithms' training principle, which states (more data yields higher accuracy). Table 4.2 shows the parameters value of the two scenarios.

Table 4.2: Parameter Value of the Two Scenarios

Parameters	Values
Channel Type	WSN
Number of nodes for 1st scenario	20, 30, 50 nodes
Number of speeds for 1st scenario	10 speed
Workspace arena	3000m x 3000m
Routing protocol	AODV
Transmission range	500m
Simulation time	300s
Connection types	Antenna, Omni directional
Attack Type	DoS (blackhole)
Type of IDS	Cooperative

Table 4.2: Parameter Value of the Two Scenarios “Tables continued”

Packet size	1024 bytes
Bandwidth	4Mbps
Number of nodes for 2nd scenario	50 nodes
Number of speeds for 2nd scenario	10,20,30,40,50 speed
Channel frequency	2.4 GHz
Traffic Type	TCP
Mobility type	Random
Traffic sending rate	32 kbps

4.2.1 The 1st Scenario Based on the Number of Nodes (20, 30, 50)

The simulation of the first scenario for MANET using the AODV routing protocol including the IDS system was executed with a different amount of nodes (20, 30,50) with a speed of 10 km/h; the network performance was recorded through a trace file from the NS2 simulation. To calculate (E2E and ARP). The measurement of performance was for the MANET network under the effect of malicious nodes (black hole) with AODV routing and IDS to show how it impacts the performance (E2E and ARP). The metrics of the MANET network are the following:

- a. E2E time delay: Time of the packet need to travel from one node to another or from sources to destination in a MANET is called E2E delay. In the proposed model, victim-attacker connectivity varies across simulation time due to mobility and node coverage limits. Victim-attacker nodes aren't always connected. Therefore, the node speed affects network performance. Hence, this delay will calculate by Eq. (4.1).

$$E2E = T_a^t - T_s^t \quad (4.1)$$

In the above equation, T is represented end-to-end, T_a^t represents the time (seconds) of reception acknowledgement, and T_s^t presents the time (seconds) when the packet is sent. The nodes with faster speeds might return to the coverage point quicker than slow-moving nodes. E2E values are evaluated after running the simulation time and obtained as average values.

- b. Average Received Packets (APR): This means each client node collects the full amount of sent messages for each sender nodes to client node. In the proposed model, the packets realized from the assault node into victim to decrease when nodes move faster, indicating that the packet has unsuccessfully been received via a station node. This average is displayed as a percentage. Mathematically can represent (total packet loss/total packet). We can calculate the average by Eq. (4.2).

$$Rx_{mean} = \sum_0^{T_{sim}} x \quad (4.2)$$

Where T_{sim} : simulation time (seconds), x: received packet counter.

The primary findings of this scenario are the E2E and ARP in the proposed network. Figure 4.1 describes the E2E delay among nodes in the network (different amount of nodes); We note the delay time increased when amount of nodes is increased to 50 amount of nodes compared to amount of nodes when it is 30 nodes.

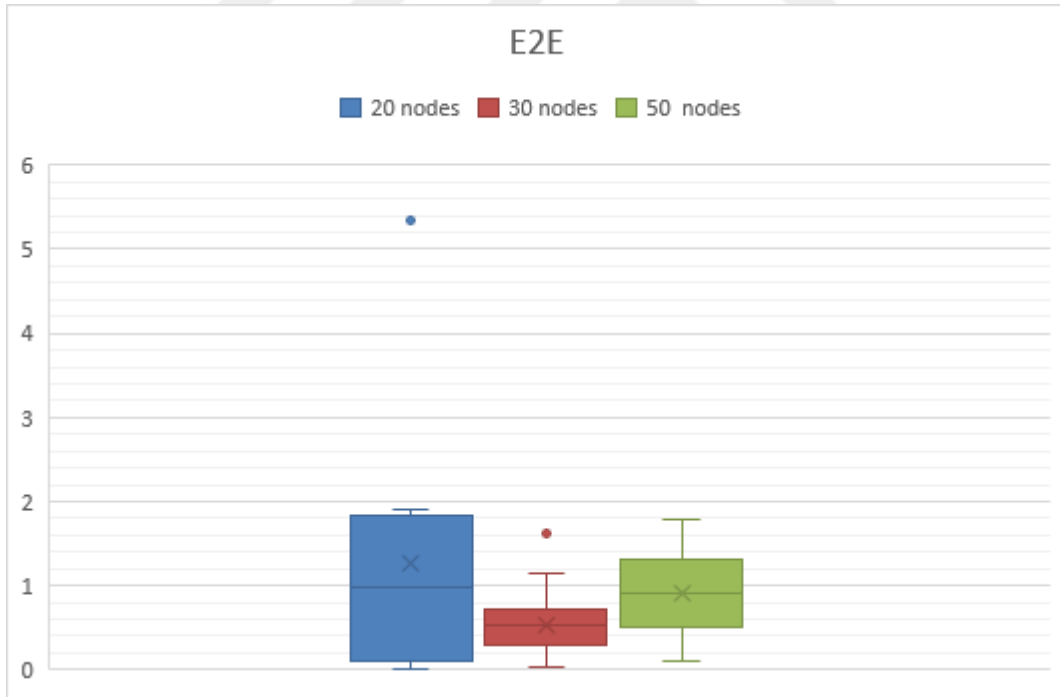


Figure 4.1: Delay Between Nodes in the Network (E2E Delay, Different No. of Nodes).

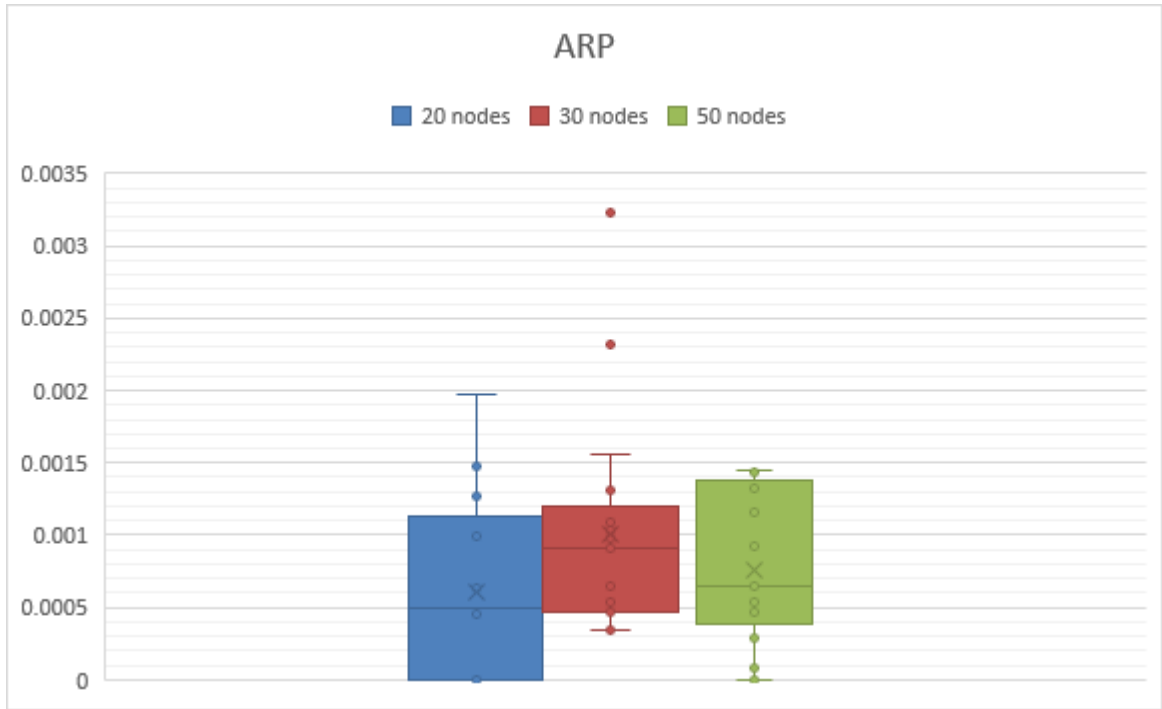
Figure 4.2 describes ARP for the MANET (different amount of nodes). We note ARP increases when the amount of nodes increases. Table 4.3 shown the effect to each amount of nodes.

Table 4.3: Average Receiving Packet in the Network With (Different Nodes)

20 nodes	30 nodes	50 nodes
0	0.001088	0.001453
0.001979	0.00156	0.001321
0.000449	0.000642	0.000692
0.000489	0.000513	0.000642
6.95E-08	0.000465	8.62E-05
0.000459	0.000479	0.000469
7.5E-08	0.000973	0.001443
0.000988	0.000539	0.000535
7.95E-08	0.001307	0.000918
0.000499	0.000394	0.001165
8.5E-08	0.001029	0.000479
0.000631	0.000346	1.86E-06
0.001477	0.000905	0.001429
0.000631	0.000346	1.86E-06
0.001477	0.000905	0.001429

Table 4.3: Average Receiving Packet in the Network With (Different Nodes) “Tables Continued”

0.00127	0.003232	0.0005
1.03E-07	0.002314	0.000295

**Figure 4.2:** Difference Between ARP in the Network With (Different No. of Nodes).

4.2.2 The 2nd Scenario Depends on Different Speeds of Nodes (10, 20, 30, 40, 50)

The simulation of the second scenario for MANET using the AODV routing protocol with the IDS system was executed and under the influence of malicious nodes (blackhole) with different speeds (10, 20, 30, 40, 50) and selected 50 nodes. The network performance was recorded through a trace file from the NS2 simulation. Then, the E2E delay between nodes and the average receiving packet (ARP) for MANET is calculated. The primary findings of this scenario are the E2E and ARP in the proposed network. Figure 4.3 describes the E2E among nodes in a MANET (different amount of speeds). We note when increasing the speed from 10 to 50, the delay time (E2E) decreases.



Figure 4.3: Delay Between Nodes in the Network (Different No. of Speeds).

Figure 4.4 describes ARP for the network (a different number of speeds). We note when increasing the speed from 10 to 50, the ARP of the network increases. This effect for each speed is shown in Table 4.4.



Figure 4.4: Difference Between ARP in the Network With (Different No. of Speeds).

Table 4.4: Average Receiving Packet in the Network with (Different Speeds)

No. of Speed=10	No. of Speed=20	No. of Speed=30	No. of Speed=40	No. of Speed=50
0.00823013	0.00823013	0.00162005	0.00162005	0.01053951
0.00477763	0.00477763	0.00203458	0.00203458	0.00280399
0.00155402	0.00155402	0.01229111	0.01229111	0.00823924
0.0129075	0.0129075	0.00764681	0.00764681	0.00706576
0.00514272	0.00514272	0.00333577	0.00333577	0.00054488
0.00236698	0.00236698	0.00191931	0.00191931	0.00227419
0.00223794	0.00223794	0.00927927	0.00927927	0.00344242
0.01156696	0.01156696	0.00394734	0.00394734	0.00253112
0.00868125	0.00868125	0.00493853	0.00493853	0.00564228
0.00462402	0.00462402	0.00170531	0.00170531	0.01114388
0.01096315	0.01096315	0.00550282	0.00550282	0.01660625
0.00692999	0.00692999	0.00283909	0.00283909	0.00252758
0.00362394	0.00362394	0.00091811	0.00091811	0.00242978
0.00840144	0.00840144	0.00942925	0.00942925	0.00146127

4.3 SUMMARY OF THE IMPLEMENTATION AND THE RESULTS FOR THE FIRST PART

As a summary of these two scenarios, we observe the amount of nodes as well as the speeds of nodes impact its performance, where (E2E delay) increase as the amount of total nodes increases; Hence, the ARP remain at a low rate, even with a system to monitor intrusion. While it was discovered when the speeds are increased, the E2E delay time and ARP increases. These results are in the presence of IDS system that monitors the MANET and gives the necessary alert to saving the MANET from attack, but it still not satisfactory for such MANET networks. For its significant uses in many applications that are important in modern life, such as its use on the IoT, which the universe views as a mechanism to facilitate users' tasks.

In general, MANET networks have become exposed to many attacks because of the node's unrestricted infrastructure for mobility. Therefore, a much more essential matter in these networks' is protected and enhanced security. Therefore, MANET requires continuous monitoring by a deep learning system. The proposed DL monitors the network's activity for signs of unusual behavior. It is possible to train the routing protocol with the IDS system of networks vulnerable to such attacks and others via machine learning algorithms to make them faster in detecting attacks and to increase protection against assaults.

In other side, the DoS attack is one of the types of vulnerability in the MANET network, an issue leading to less protection of the network. So, the monitoring of the activity of the MANET network of the regular or malicious anomalous base using IDS. In other words, most threshold-based intrusion detection systems use an activity threshold or less to identify abnormal invasions and conditions above the threshold that have been classified as intrusive. The ability to perceive previously unknown attacks is the most significant benefit of the anomalous analysis system, but this system sometimes fails to detect more well-known security attacks.

The second part includes the results of the data depending on 50 nodes and different speeds (10, 20, 30, 40, 50) as a dataset that was adopted in training the deep learning (DL) algorithms. Using MATLAB code, simulations, and training techniques have been carried out to generate ARP and E2E delay estimations at the prompts. This issue could be addressed with some MATLAB code. Therefore, the dataset was modified by multiplying the data by the matrix to get a shape that can train using deep learning algorithms (CPBNN, FFNN, CNN), which had previously been just random data. The DL using nodes and characteristics of MANET to training the data from MANET nodes as well as characteristics allow training and decode the packet. Hence, packets in a big network can be evaluated for their integrity.



4.4 THE SECOND PART, THE TRAINING OF DATA FOR THE PROPOSED NETWORK BY USING DEEP LEARNING ALGORITHMS

The proposed method is implemented using the different artificial neural network (ANN) models employed for deep learning. These algorithms (CBPNN, FFNN and CNN) train by MATLAB. The configurations of ANN to train the data in the proposed network are in Table 4.5, and as we mentioned earlier, this data has been processed in order to be suitable for training. Therefore, Figure 4.5 shows the data set after processing.

Table 4.5: Configurations of Ann.

Particle	Details
No. of hidden layers	2 hidden layers
Training method	SL / DL
Epochs Number	100
Max gradient	1 e (-30)
MSE	1 e (-20)
ANN types (respectively)	CBPNN, FFNN, CNN
Validation data	K-fold cross-validation partition
No. of Test Sets	10 set

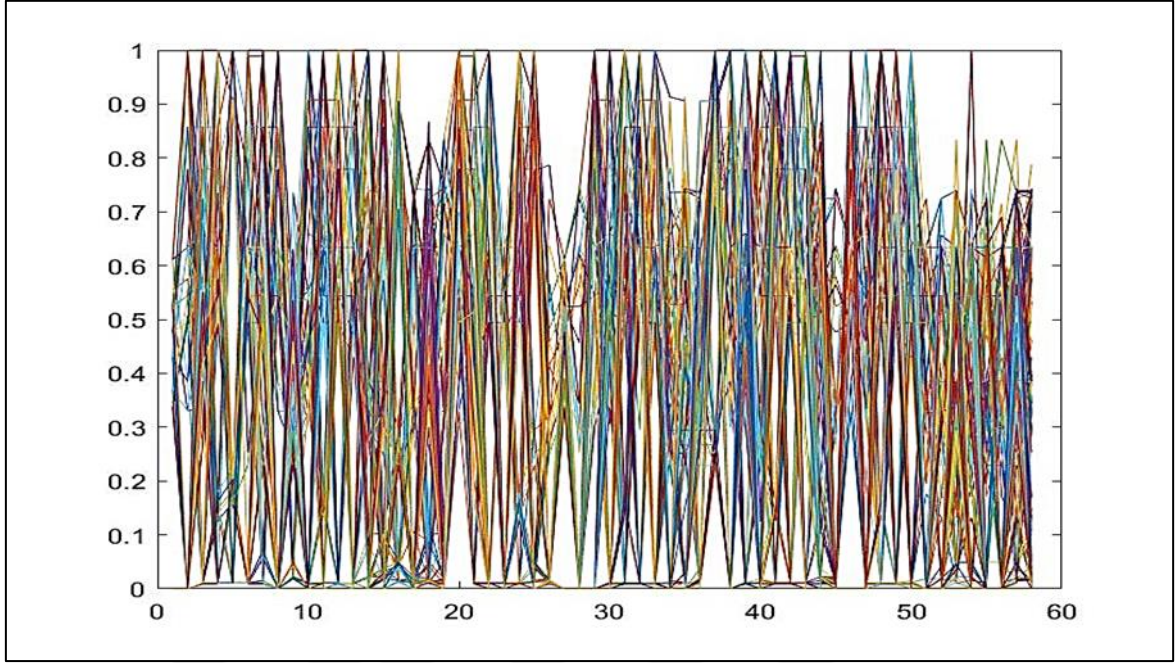


Figure 4.5: Proposed Network After Processing Data.

4.4.1 Result for the 1st Algorithm CBPNN

After the data is processed and then trained by embedding it in the DL models, a design of the (CBPNN) algorithm is displayed in Figure 4.6. by using MATLAB-GUI. Nodes are detection in a random architecture, with 4 nodes effecting as (the black hole) nodes of assaults. This design is for the network with a one input layer (500 neurons), two invisible layers (10 sets), and a one output layer. The performance of CBPNN is presented in Table 4.6 display (Accuracy, MSE, MAE, RMSE and other parameters). The MANET network used parameters to build (ANN) inserted above (in Table 4.5). To determine a performance of a MANET. The results of this model, like accuracy of detecting (black hole) assaults and other metrics as (MSE, MAE and RMSE) for the CBPNN algorithm, is presented in Figure 4.7.

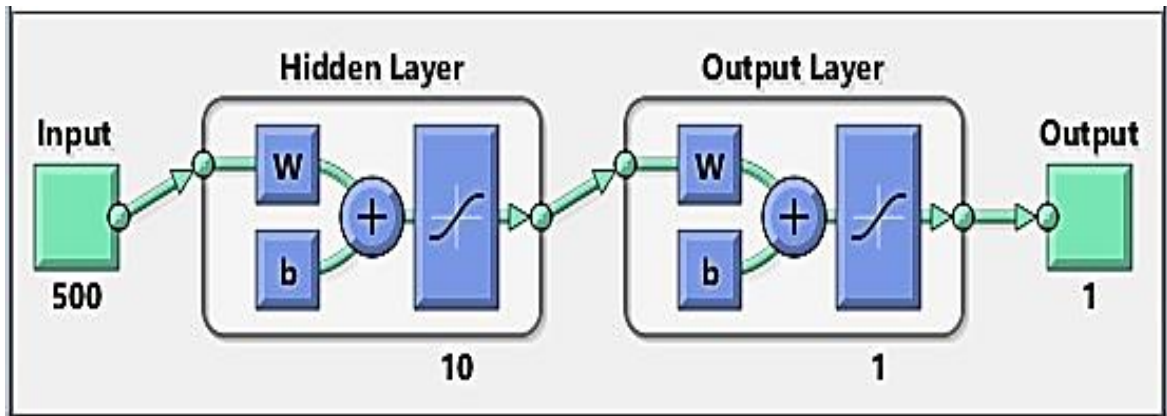


Figure 4.6: Design of the Cbpnn Algorithm.

Table 4.6: Performance of Cbpnn Algorithm.

Metrics	Results
Validation data	K-fold cross-validation partition
Number of Observations	58
Number of Test Sets	10
Accuracy	74 %
MSE	1.4138
MAE	0.5172
RMSE	1.1890
Run time	27s from 60 sec

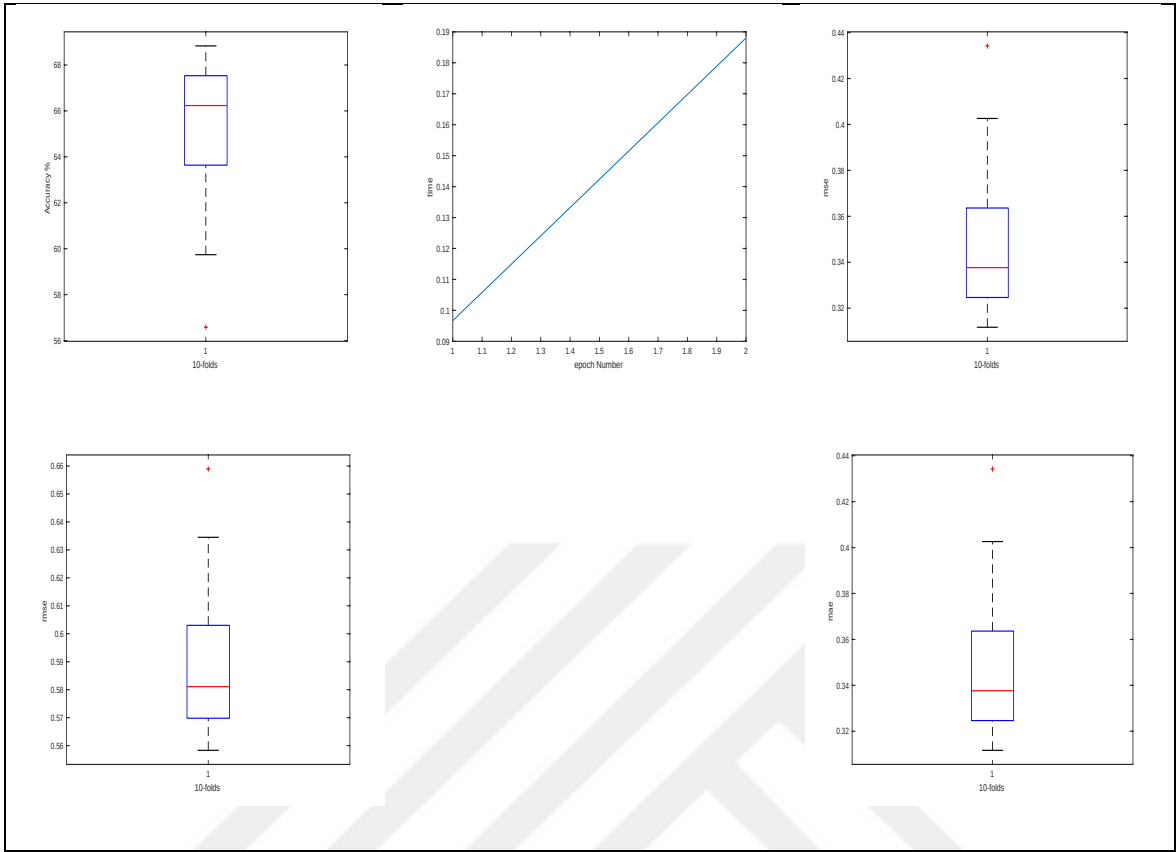


Figure 4.7: Accuracy of Detecting a Black Hole Attack of the Cbpnn Algorithm.

4.4.2 Result for the 2nd Algorithm FFNN

The design of the second (FFNN) algorithm is displayed in Figure 4.8. by using MATLAB-GUI, also, after the data is processed and then trained by embedding it in the DL algorithm, Nodes are detected in a random architecture, with 4 nodes effecting as (the black hole) nodes of assaults. This design is for the network with a one input layer (500 neurons), two invisible layers (10 sets), and a one output layer. The performance of FFNN is presented in Table 4.7 display (Accuracy, MSE, MAE, RMSE and other parameters). The MANET network used parameters to build (ANN) inserted above (in Table 4.5). To determine a performance of a MANET. The results of this model, like accuracy of detecting (black hole) assaults and other metrics as (MSE, MAE and RMSE) for the FFNN algorithm, is presented in Figure 4.9.

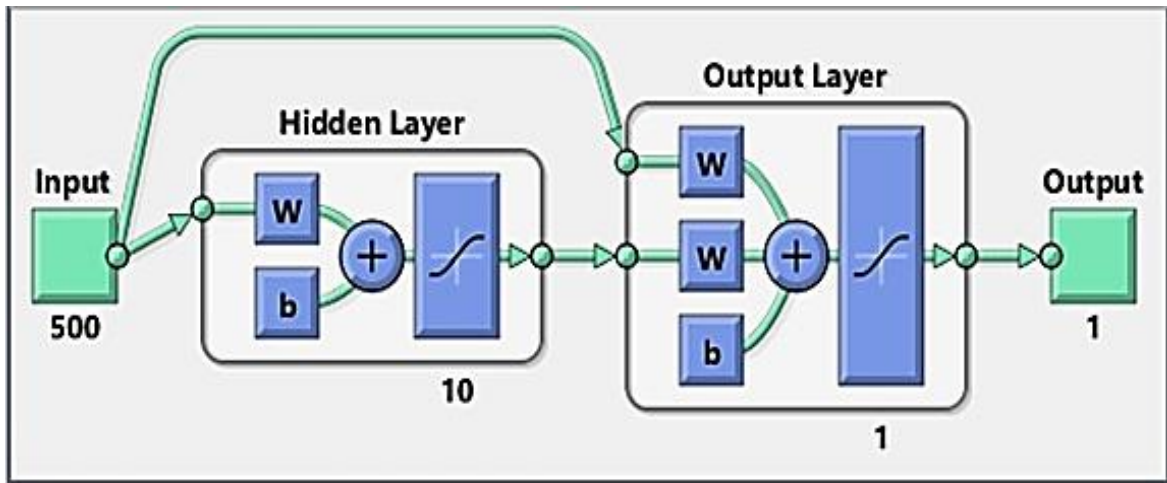


Figure 4.8: Design of the Ffnn Algorithm.

Table 4.7: Performance of Ffnn Algorithm.

Metrics	Results
Validation data	K-fold cross-validation partition
Number of Observations	50
Number of Test Sets	10
Accuracy	82 %
MSE	0.7069
MAE	0.3276
RMSE	0.840
Run time	18 s from 60 sec

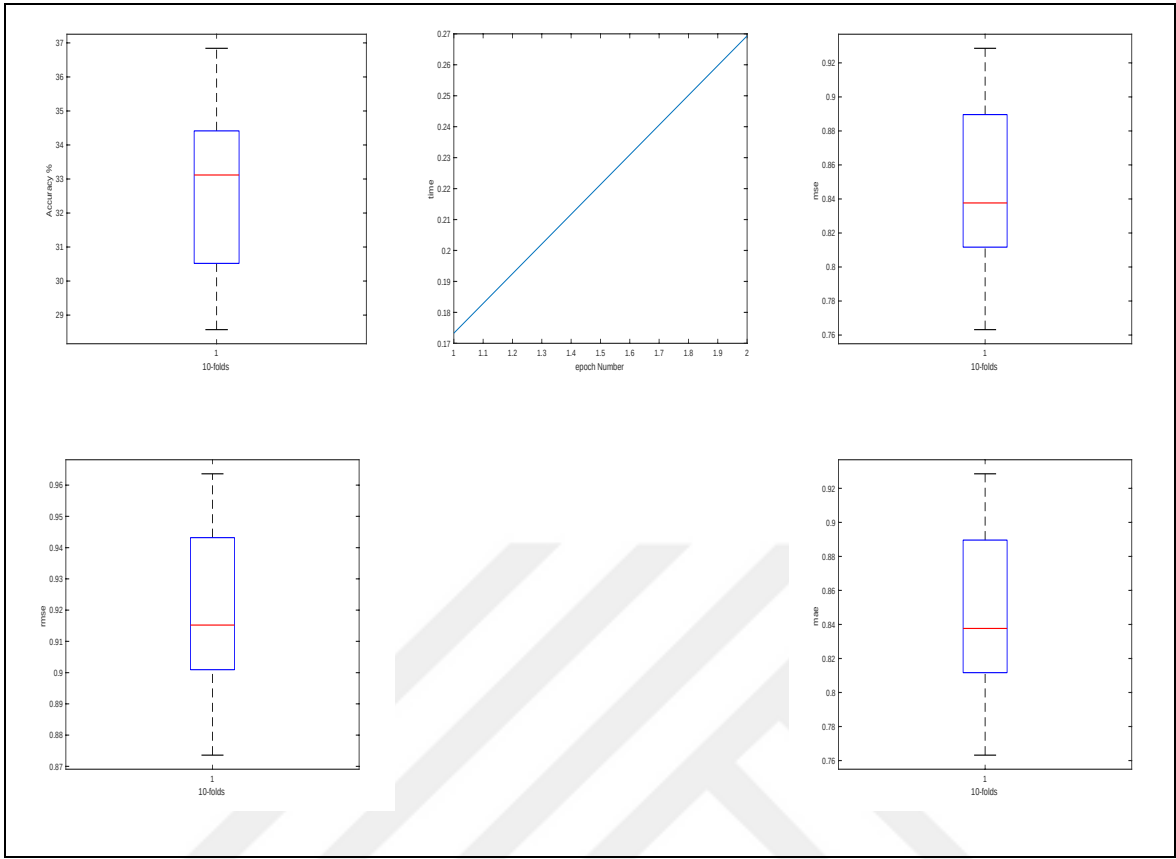


Figure 4.9: Accuracy of Detecting a Black Hole Attack of the Ffnn Algorithm.

4.4.3 Result for the 3rd Algorithm CNN

The third model (CNN) algorithm is displayed in Figure 4.10. by using MATLAB-GUI, as a similar of two algorithm mean after the data is processed and then trained by embedding it in the DL algorithm, Nodes are detection in a random architecture, with 4 nodes effecting as (the black hole) nodes of assaults. This design is for the network with a one input layer (500 neurons), two invisible layers (10 sets), and a one output layer. The performance of CNN is presented in Table 4.8 display (Accuracy, MSE, MAE, RMSE and other parameters). The MANET network used parameters to build t(ANN) inserted above (in Table 4.5). To determine the performance of a MANET. The results of this model, like accuracy of detecting (black hole) assaults and other metrics as (MSE, MAE and RMSE) for the CNN algorithm, is presented in Figure 4.11.

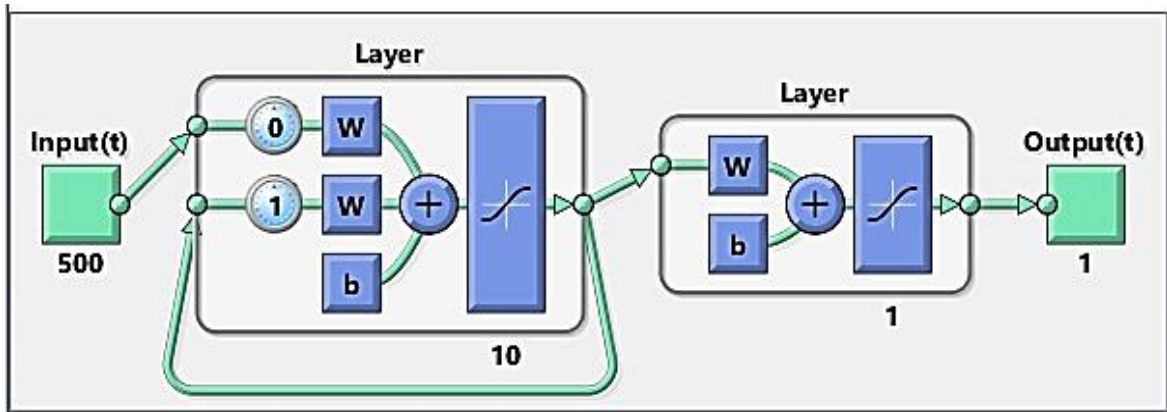


Figure 4.10: Design of the CNN Algorithm.

Table 4.8: Performance of CNN Algorithm.

Metrics	Results
Validation data	K-fold cross-validation partition
Number of Observations	50
Number of Test Sets	10
Accuracy	85 %
MSE	0.9828
MAE	0.3276
RMSE	0.9913
Run time	17 s from 60 sec

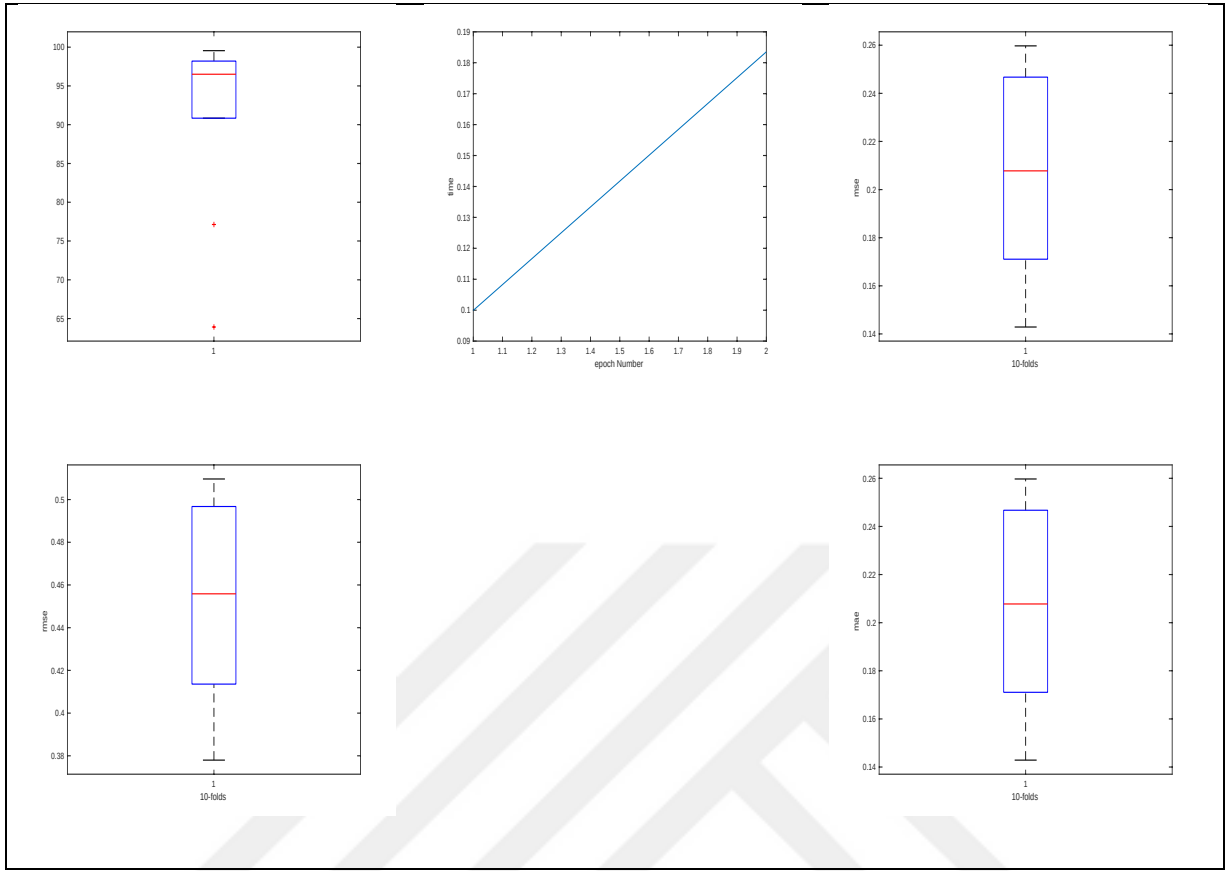


Figure 4.11: Accuracy of Detecting a Black Hole Attack of the CNN Algorithm.

4.5 SUMMARY OF TRAINING AND BEST RESULT OF THE ALGORITHMS

When implementing the training and testing the proposed network data of each algorithm, Figure 4.12 displays the results of the three model (CPBNN, FFNN and CNN) algorithms, respectively, depending on the gradient value of 1, which represents the threshold value. The best result achieved CNN algorithm achieved the best detection of (black-hole) assault, depending on the accuracy of the detection attack and time of detection. Where the detection accuracy reached 85 percent in a time of 17 seconds, it is the best accuracy achieved by the algorithm (CNN) between the three algorithms.

As well as Figure 4.13 explains precise details of the best algorithm (CNN) described by the weight of gradient equal 1.1567, Mu equal 0.001, and validation checks equal 8 positions on the epoch value equal 10. In contrast, other positions (see positions 0 and 2) these positions do not check validate of the algorithm (CNN).

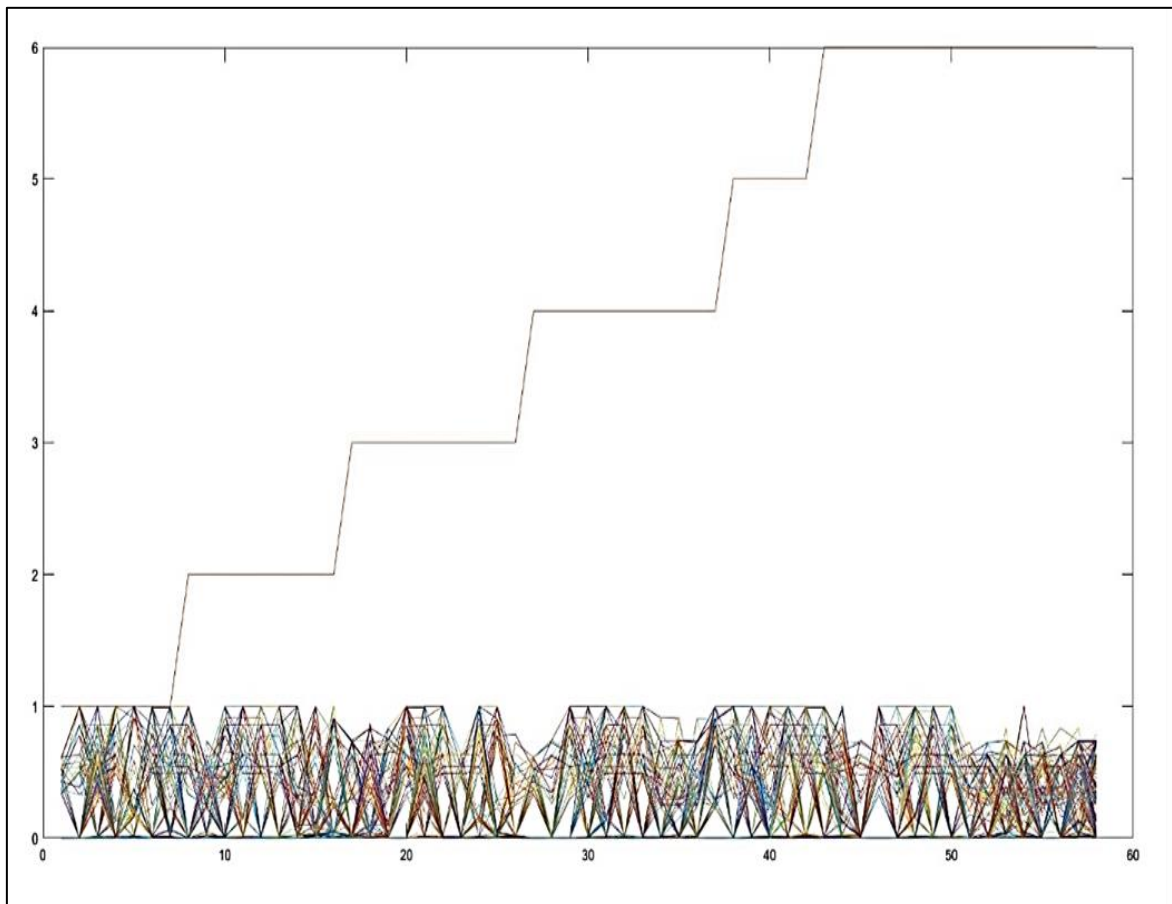


Figure 4.12: Results of the Three Algorithms Cpbnn, Ffnn and CNN (Matlab Results).

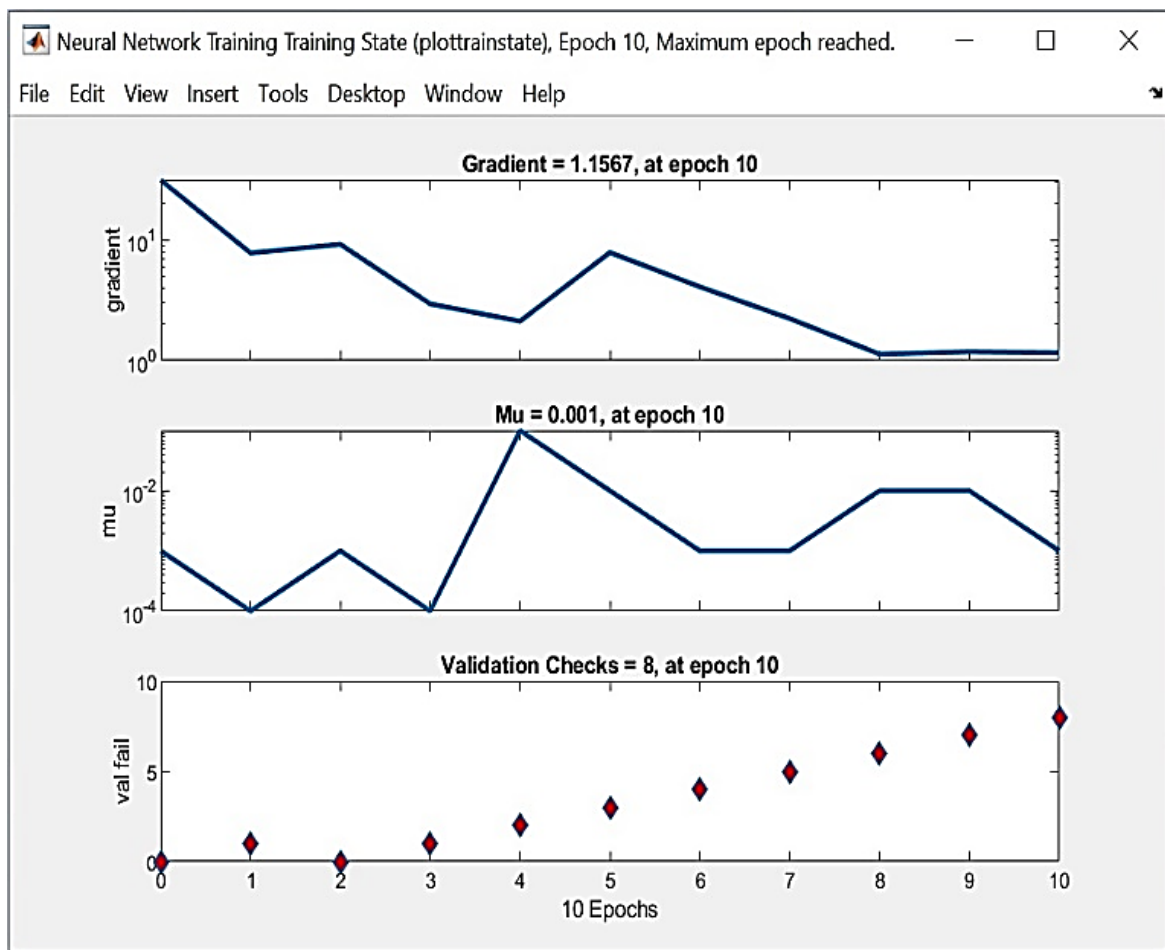


Figure 4.13: Result for the Best Algorithm Cnn (Matlab Results).

Relying on mean square error (MSE), Figure 4.14 shows the best validation performance is 2.8118 at 2 epochs; this indicates the best detection speed of verification, in addition to showing both the train line and the set line and the best position for the best goal. This score appeared when implementing (CNN) algorithm. In addition, Figure 4.15 shows the detailed training tape with the design for the best model (CNN) algorithm using MATLAB.

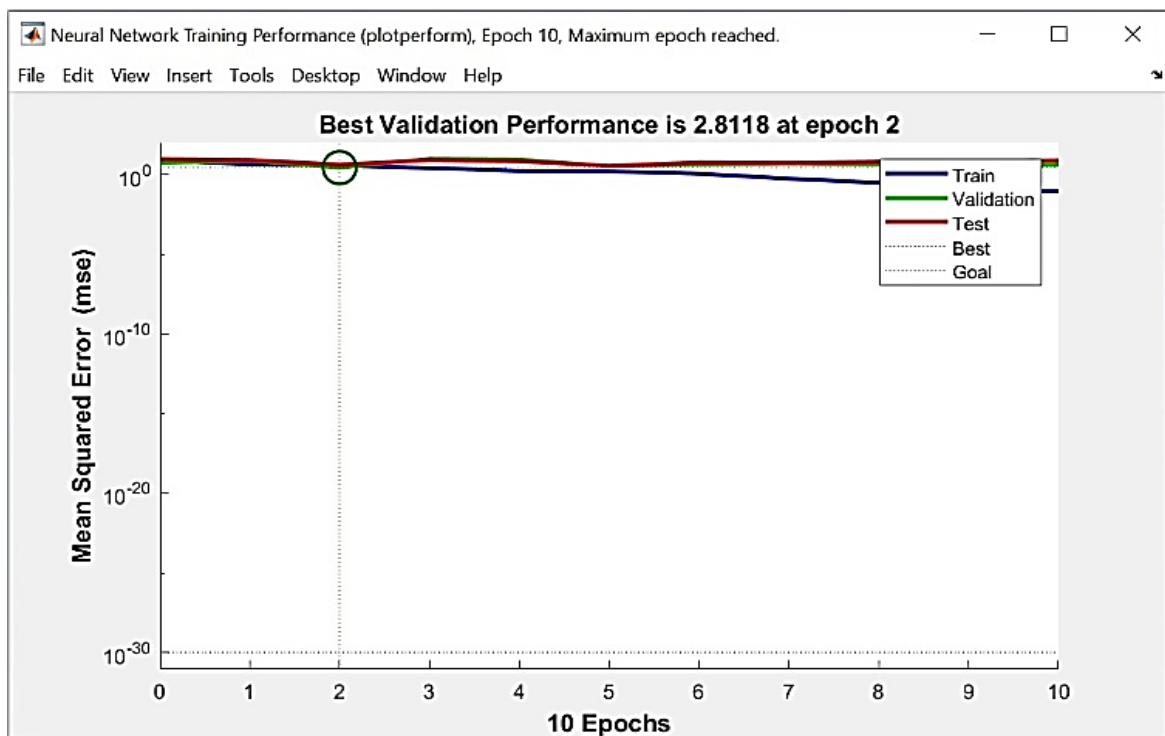


Figure 4.14: Best Validation Goal for the Performance of the Algorithm (Matlab Result).

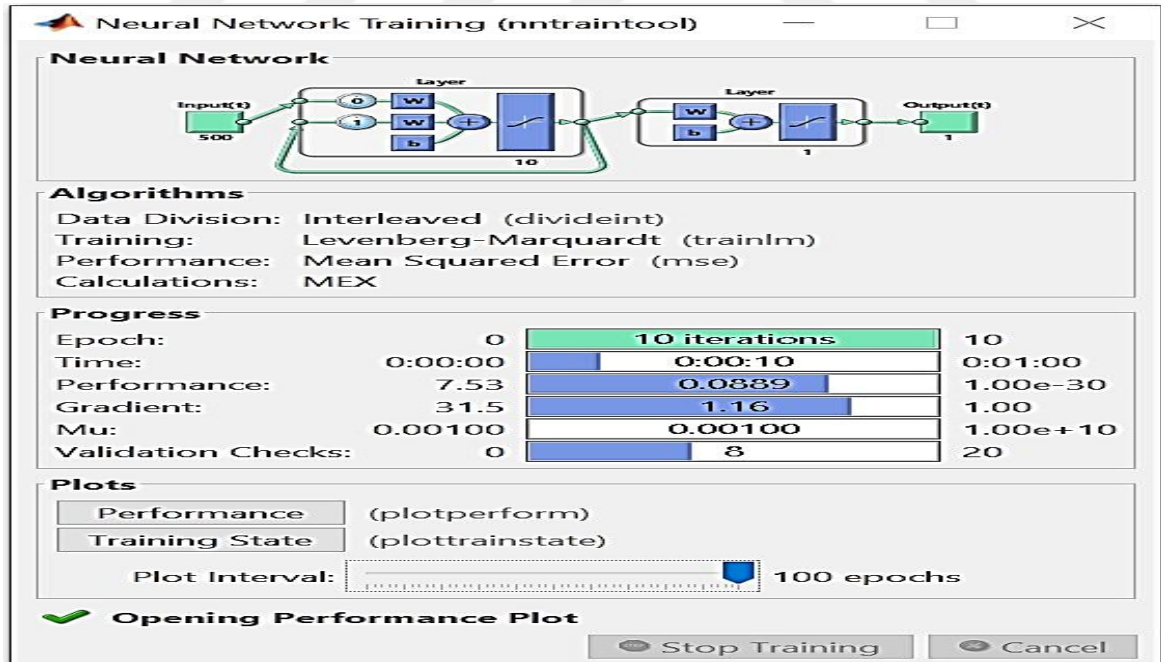


Figure 4.15: Detailed Training Tape with the Design for the Best Model (Matlab Result of Cnn Algorithm).

In summary, we found the results of the implementation for the proposed model using deep learning algorithms (CBPNN, FFNN and CNN), respectively. The accuracy achieved is (74%, 82% and 85%), at the time (27 s, 18 s, and 17 s) from 60 seconds. Utilizing 10 values of every single set, (10-K fold cross-validation) was employed to calculate the effectiveness of the algorithm.

Additionally, metrics (MSE, MAE, and RMSE) are calculated depending on each algorithm's accuracy, as shown in Table 4.9. Based on the outputs of the first algorithm (CBPNN), we observe the CBPNN achieve less accuracy than other algorithms. While the outputs of the second algorithm (FFNN) achieved better accuracy than the previous algorithm (CBPNN) and less than the CNN algorithm because the outputs of the CNN algorithm achieved the best accuracy among the three algorithms.

Table 4.9: Performance of the Metrics (MSE, MAE, and RMSE) for Each Algorithm.

No of Nods	Algorithm	Acc	MSE	MAE	RMSE	Time From 60s
50	CBPNN	74.1379	1.4138	0.5172	1.1890	27s
50	FFNN	82.7586	0.7069	0.3276	0.8408	18s
50	CNN	85.7586	0.9828	0.3276	0.9913	17s

To describe the difference among these metrics for the implemented three algorithms by deep learning (DL). Figure 4.16 illustrates the difference for each algorithm's accuracy, times, and metrics to detect a black hole attack. The metrics mean square error (MSE) impacts the accuracy of detection, where the closer this value is to one, based on the established threshold, the more accurate the detection of a black hole attack. Consequently, the impact on the average receiving packet (ARP).

While the mean average error (MAE) impacts the time of detecting a black hole attack, hence, impacting on sending of data from one node to another, in other words, impacting end-to-end (E2E) delay, as explained in the next chapter.

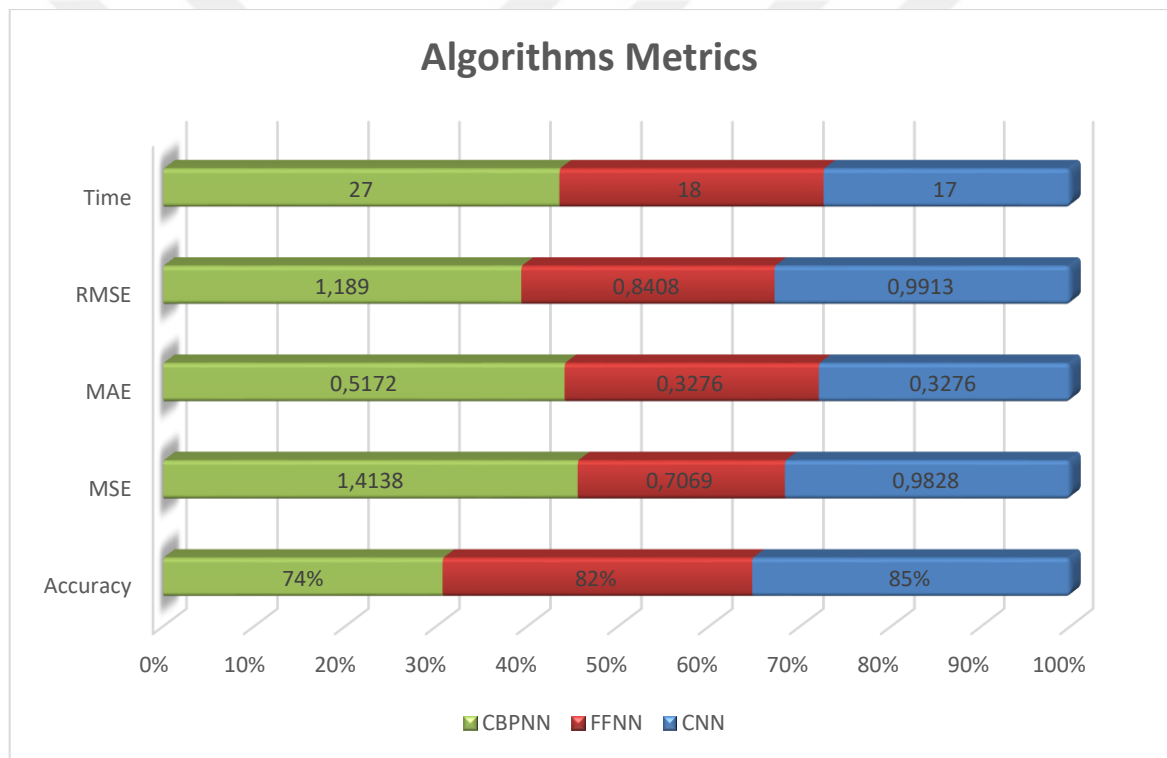


Figure 4.16: Differences Among Three Algorithms to Detect Manet Attack.

5. RESULTS AND DISCUSSION

5.1 INTRODUCTION

To optimize network resource utilization and detect attacks (blackhole) on MANET networks. This chapter will analyze the results and compare them with three comprehensive comparisons. The first comparison shows the difference between algorithms trained by deep learning (DL). To determine which is better, depending on detection accuracy, detection time, and the effect on network performance. Accordingly, we've included network criteria (ARP and E2E) to make comparisons easier. This comparison was made among algorithms (CBPNN, FFNN and CNN) accuracy in terms of the effect on the (ARP and E2E delay) for a proposed model to measure the MANET performance. While the second comparison explains the effect of deep learning (DL) on MANET networks before and after training the intrusion detection system by (CBPNN, FFNN, and CNN). In addition, the third comparison illustrates the strengths and weaknesses between the best algorithms (CNN) and the algorithms of other studies that used the IDS system to determine the accuracy of detecting the attack on MANET. This comparison was made using the same dataset (KDDTest-21) [77] used in other studies to be a fair comparison. This comparison will be comprehensive for deep learning algorithms that address prevalent difficulties in MANETs Networks from 2015 to 2021, such as difficulty detecting attacks against the MANETs.

5.2 COMPARISON OF THE DIFFERENCE AMONG (CBPNN, FFNN AND CNN) ALGORITHMS AFTER TRAINING MANETS.

The results collected from the experiments conducted in this work are summarized in Table 5.1 using DL for MANET. By analyzing its strengths and shortcomings, select which algorithm is best suited for the task. The creator of MANETs could use this as a guide to develop the DL solution tailored to their individual requirements. This comparison shows that the proposed method enhanced the network's performance after training. Where using the CBPNN algorithm has achieved low detection of (blackhole) attacks. That means affecting the network's performance metric, where increasing the average receiving packets (ARP) but less than (FFNN and CNN) and decreasing end-to-end (E2E) delay but also less than (FFNN and CNN).

In contrast, The FFNN algorithm achieved more accuracy in detecting (blackhole) attacks than the CPBNN algorithm, where it enhances the average receiving packets and end to end delay (ARP and E2E delay) but less than the CNN. Depending on a training of the MANET network data. Whereas the CNN algorithm achieved the best accuracy in detecting (blackhole) attacks among proposed algorithms, it more enhancing to the average receiving packets and end to end delay (ARP and E2E delay). Depending on a training of the MANET network data. Figure 5.1 are illustrated these practical solutions using deep learning.

Table 5.1: Comparison of the Various Effects of Accuracy on ARP and E2E Delay Based on Each Algorithm's Performance Metrics.

No of nods	Algorithm	Accuracy %	MSE (ARP)	MAE (E2E)	RMSE	Time From The 60s
50	CBPNN	74.1379	1.4138 (Good)	0.5172 (Good)	1.1890	27s
50	FFNN	82.7586	0.7069 (Better)	0.3276 (Better)	0.8408	18s
50	CNN	85.7586	0.9828 (Best)	0.3276 (Best)	0.9913	17s

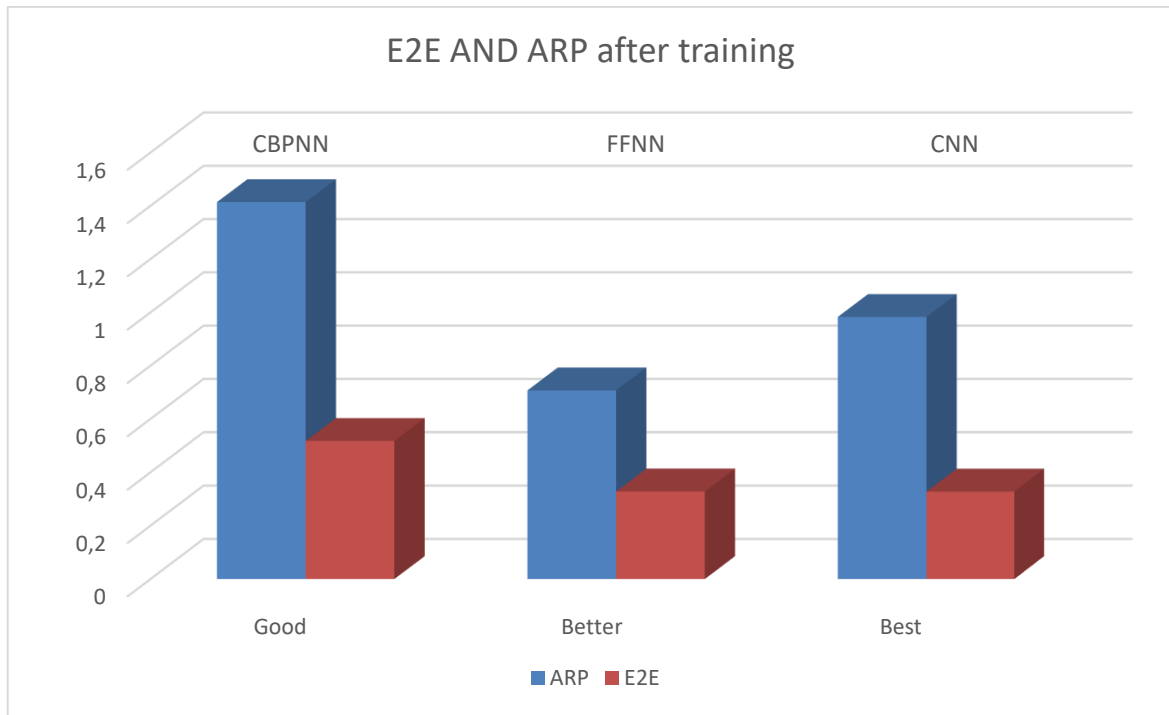


Figure 5.1: Manet Network Performance After Training Data by Each Algorithm.

We note in the figure above the enhancement in network performance after training the proposed network through the improvement of the IDS system using DL algorithms to recognition (blackhole). Hence, by these results, our study reaches the desired goal using DL for the IDS system to work with more reliability and protect MANETs networks from attacks that are exposed to and affect the performance.

5.3 COMPARISON OF THE NETWORK PERFORMANCE BEFORE AND AFTER TRAINING

In this comparison, we will explain the difference between the performance of the proposed network before and after training with deep learning algorithms (CPBNN, FFNN, and CNN) by calculating (ARP and E2E) for the network. Figure 5.2 shows a significant decrease E2E delay before and after training the IDS system by the algorithms of DL , and Figure 5.3 shows an increase in the average receiving packet (APR) before and after training the IDS system by using algorithms of DL.

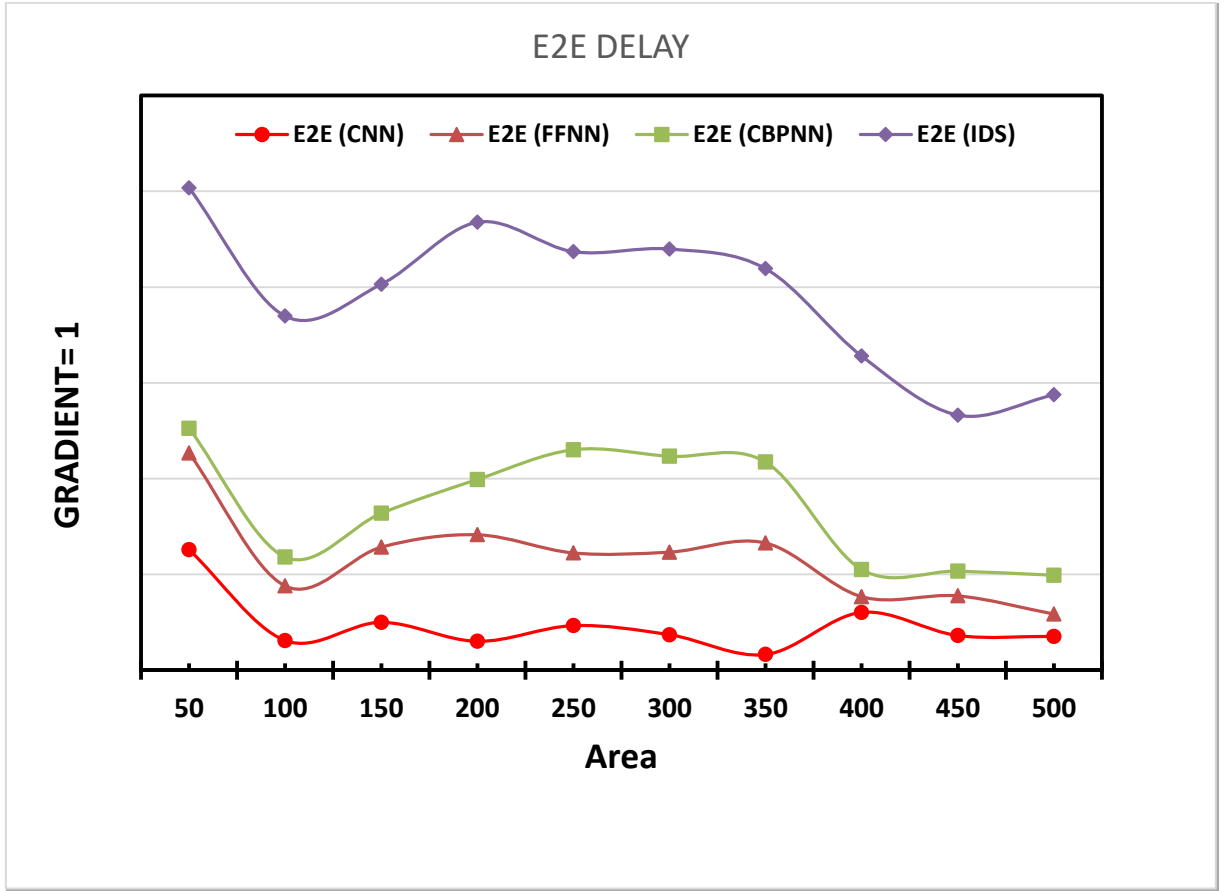


Figure 5.2: Difference Between E2E Before and After Training by Proposed Algorithms.

This difference exhibits the deep learning capability of improving network performance by decreasing the end-to-end (E2E) delay by detecting (black hole) attacks using training the intrusion detection system (IDS) in deep learning algorithms. However, each algorithm's accuracy and time to detect (DoS) attacks differ. Where each algorithm affects the improvement of this decrease to a certain level, it was noticed that the first algorithm (CBPNN) reduced this delay between nodes at a lower rate than other algorithms (FFNN and CNN), which means the improvement with a reduced delay is good, hence, increasing the network performance and protecting it against the attacks that networks are exposed to MANET. While the second algorithm (FFNN) enhanced the performance of the proposed network more than the first algorithm (CBPNN) by lowering the delay between nodes. In contrast, the third algorithm (CNN) enhanced the network's performance to the greatest extent feasible by minimizing the delay between nodes the most, i.e., better than the two algorithms (CBPNN and FFNN).

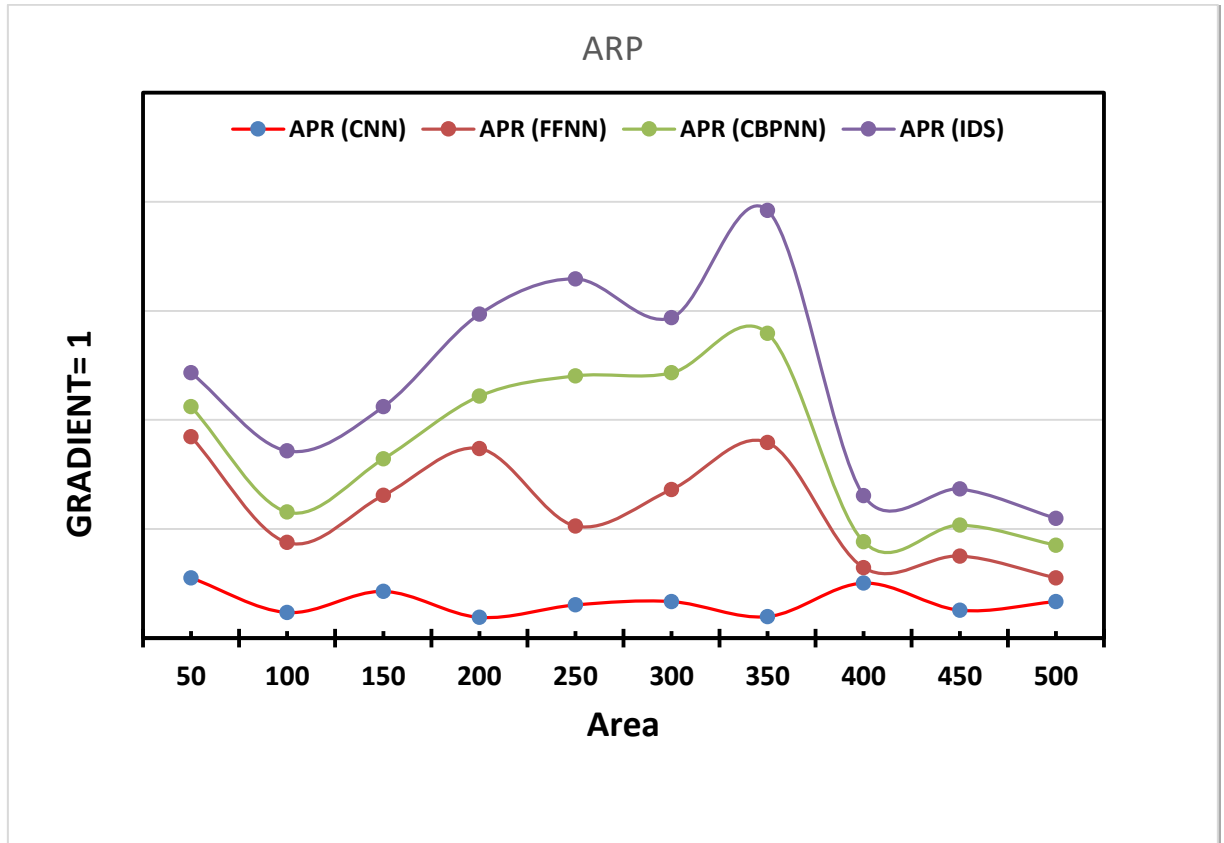


Figure 5.3: Difference Between ARP Before and After Training by Proposed Algorithms.

Likewise, for the average receiving packet (ARP), DL has a significant impact on improving MANET performance. Through increasing the average receiving packet (ARP) by detecting (black hole) attacks and using training the IDS system in DL algorithms. In consideration of the differences among algorithms proposed to improve network performance. It was noticed that the first algorithm (CBPNN) increased the ARP at a lower rate than other algorithms (FFNN and CNN). That means the improvement with an increased ARP is the excellent performance of the network. Hence, increasing the network performance and protecting it against the attacks that networks are exposed to MANET.

In contrast, the FFNN algorithm enhanced the performance of the proposed network more than the CBPNN by higher the ARP from the S node to the D node (SN to DN). While the CNN algorithms enhanced the network's performance to the greatest extent feasible by maximizing the ARP from the S node to the D node (SN to DN), i.e., better than other algorithms (CBPNN and FFNN).

5.4 COMPARISON BETWEEN THE PROPOSED MODEL AND OTHER STUDIES

The remarkable technological advancements over the previous decade have enhanced every aspect of living. Additionally, there is the issue of illegal data, necessitating installing an effectiveness of the IDS system. The "curse of lower accuracy," however, has been demonstrated through previous studies to cause low detection rates, longer detection times, and lower accuracy when dealing with unbalanced network data. Therefore, in this comparison, we used the best algorithm (CNN) to train a data set (KDDTest-21) [77] used in previous studies. To analyze the metrics of algorithms and their effect on network performance (E2E and ARP) within the same environment.

To implement this comparison, we need to reflect the existence of a black hole attack; therefore, the input parameters are chosen based on pre-processing of originaive KDDTest-21 dataset is essential to make it an appropriate input for learning method. We necessitate to transubstantiate a nominal features to numeral values. Only column No.2 represent (type of protocol), column No.3 represent (services), column No.4 represent (flag), and column No.42 represent (detection or Non) contains nominal values. The experiment is performed on a PC same PC that implement previous three models of deep learning (Chapter 4). This designed to performance of the CNN model for detection attack such as, DoS.

The amount of input nodes will be specified from an input data set. The amount of nodes in hidden layers in the neural network are varied frequently during the experiment to achieve a highly accurate and stable neural network model and to avoid any overfitting.

5.4.1 Evaluation Metrics for Comparative Study

Our model evaluates the effectiveness of the CNN-IDS system using the most widely-used performance metric for IDS, like Accuracy (ACC). We also include (Detect Rate-DR) as well as (False Positive Rate-FPR). These measurements are extracted as follows:

- a. True-Positive or TP: refers to the amount of abnormality records that have been correctly detected like, which is the same as the amount of records that have been correctly rejected.
- b. False-Positive or FP: refers to the erroneously rejected and represents the amount of normal records that have been detected as regular.
- c. True-Negative or TN: refers to the correctly admitted and represents the amount of normal records identified as regular.
- d. False-Negative or FN: refers to the wrongly admitted and represents the amount of anomaly records classified as regular.

Hence,

Accuracy: represent percentage of the amount of records classified correctly against total a records, the accuracy calculated by in (Eq 5.1).

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \times 100\% \quad (5.1)$$

True Positive Rate (TPR): refers to the Detection Rate (DR), it displays percentage of the amount of records detected correctly across a total amount of anomaly records, the DR calculated by in (Eq 5.2).

$$DR = \frac{TP}{TP+FN} \times 100\% \quad (5.2)$$

False Positive Rate (FPR): refers to percentage of the amount of records disallowed incorrectly is distributed via a total amount of normal records, the FPR calculated by in (Eq. 5.3).

$$FPR = \frac{FP}{FP+TN} \times 100\% \quad (5.3)$$

As a result, the objective of the IDS is to achieve a greater detection rate (DR) and accuracy (ACC) with a reduced false positive rate (FPR).

5.4.2 Comparative Study of Four DL Models-Based IDS System with Proposed

Model

With a view to compare together with other deep learning models, disparity experiments are designed at same environment. In the experiment for detection, We have evaluated the effectiveness of proposed model (CNN) with four algorithms (CNN, RNN, BLSTM, DBM), as mentioned in [78]-[80]. In a similar vein, we analyze of a proposed model (CNN) depended on the (KDDTest-21) dataset, our model acquires the best accuracy, DR and same FPR with model RNN by [78]. When there are (100 node) in each hidden layer since in, each hidden layer (10 set), in each set include (10 node), and the train ratio is 0.8. Based on these processing, Table 5.2 shows the compare between the CNN-IDS on the testing set KDDTest-21 with other studies.

Table 5.2: Comparison of the CNN Algorithms with Other Studies Using KDDTest-21 Dataset.

Researchers	Title of paper	Algorithm	Type of attack	Acc %	FPR	DR	Year
Research [78]	A DL Approach for IDS using RNN	RNN	DoS	68%	2.0%	83%	2017
Research [79]	The review & comparative analysis of ML techniques for IDS in MANET	BLSTM DBM	DoS	75% 66%	19% 22%	67% 54%	2020
Research [80]	A DL Methods for IDS based ML in MANET	CNN	Dos	77%	16%	80%	2019
Proposed Method*	/	CNN	DoS	82%	2.0%	99%	2022

The experiments show that the proposed model (CNN-IDS) worked with a better detection rate (99% DR) when given 10 epochs for the KDDTrain-21 dataset and achieved accuracy reach to 82% ACC. In [78], the authors have shown the results obtained by RNN model 68% accuracy and detection rate (83% DR). While the BLSTM and DBM algorithms also gives 75% and 66% accuracy and detection rate (67% and 54% DR) [79].

Finally, other authors using the CNN model and obtained 77% accuracy and detection rate (80% DR) [80]. Which is the most recent literature about the algorithms of ANN algorithms to the field of IDS. Propitiously, all of these results are based on the KDDTest-21 benchmark dataset. Figure 5.4 demonstrates that the CNN-IDS model performs significantly better than other classification algorithms.

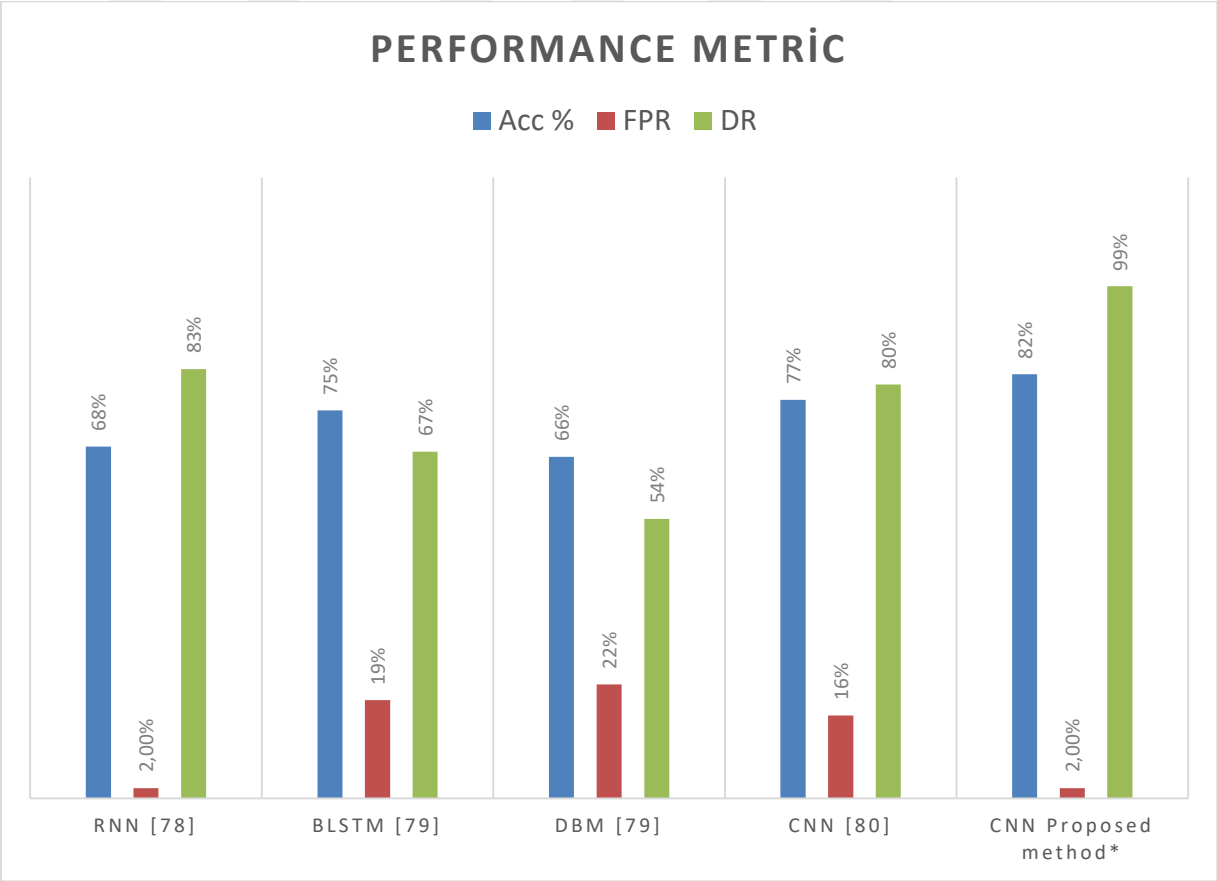


Figure 5.4: Performance Metric of the Proposed Model CNN With Other Studies to Detection of the Attack.

6. CONCLUSIONS

The MANET is known as Mobile Ad-hoc Networks complicated into multi-hop infrastructure. In other words, Due to the random organization, nodes in a MANET environment are unrestrained in their freedom to join or leave at any time and to engage in bidirectional communication. Therefore, MANETs are distinct because they can accommodate any network topology as well as any amount of nodes. This unrestricted mobility of network nodes raises major security problems because the paths set up for data transmission are not steady or not dependable, which could be problematic for time-critical applications where privacy is paramount. On the other hand, IoT's growth, Scalability, and cost factor, which are beneficial to many industries such as education, healthcare, and others, are leading many devices connected every second to expand—continuously, making it challenging to allocate resources and services.

Moreover, the movable, dynamic nodes in this architecture can be more susceptible to attacks, particularly in crucial locations such as attacks zone telecommunications where security is unavoidable. Depending on the network topology (random) and attacks that caused packet loss, made identifying virulent nodes that can inject fake routes or lose packets owing to attacks, like (black hole) attacks is difficult. Consequently, choosing a robust and dependable system that can block harmful actions and protect the network is essential.

This thesis examines the impact of the attacks on the network, E2E delay and ARP. A deep learning is utilized to identify network attacks and deliver packets along the transmission process. Where enhancing ARP and E2E delay. The guide also included to resolving the congestion of wireless mobile node network.

Therefore, a first goal in this work is for Deep learning (DL) is using to recognize attacker nodes via using the IDS system and training it depending on deep learning (DL), which can identify malicious nodes in MANETs. Where deep learning (DL) the MANETs assist future predictions of security concerns. It also provides the ability to learn by the statistical and logical procedures MANETs, stimulating the networks to adapt to new environments.

The second goal of detect complex patterns and create a secure MANET network by concentrating on security features via distinguishing virulent nodes and relieving attacks. The suggested model used deep learning to determine which neighbor node in a MANET is the most trustworthy and reliable for sending and receiving data.

The intrusion detection system will assist by deep learning and examine the MANET data sample to detect attacks. Intrusion detection systems (IDSs) take DL methodology, including high accuracy and the ability to identify or classify assaults independent of the surrounding environment. This work delves into the specifics of deep learning-based intrusion detection techniques, which could prove helpful in appropriate approaches in MANETs.

The fundamental objective of this work, the selection of a dedicated node for transference, has been accomplished by detecting malicious nodes and their differentiation from ordinary nodes and establishing a safe network architecture; by using the three algorithms. The implemented algorithms are (CBPNN), and FFNN, as well as CNN). These three algorithms use deep neural networks (DNN) to build designs to enhance the performance of IDS systems, which means IDS acts in conjunction with deep learning (DL). For what provides deep learning (DL) techniques from statistics and logical methods in the MANET network learnedness capabilities and encourages adaptation to different environments.

Finally, embedded network utilization is explained broadly in this project; reviewed a techniques of machine learning (ML) applied to in-network training and analyzed the study results. In addition, we compared the proposed algorithms with each other and with other studies. After analyzing the results of each method, the comparison of the proposed algorithms yielded a CBPNN accuracy of 74% in 27 seconds, FFNN accuracy of 82% in 18 seconds, and CNN accuracy of 85% in 17 seconds. Depending on the results provided, the best result is achieved by the CNN algorithm, which leads to a better average receiving packet (ARP) and end-to-end (E2E) decrease. At the same time, comparing the proposed model with other studies showed that the result of the CNN algorithm is better. It achieved 82% accuracy and 99% detection rate, which means better than the CNN implemented by [78], which reached 68% accuracy and 83% DR when both KDDTest-21 dataset were exposed to the DoS.

As well, the BLSTM and DBM algorithms also gives 75% and 66% accuracy and detection rate (67% and 54% DR) [79]. Finally, other authors using the same model (CNN) and obtained 77% accuracy and 80% DR [80].

Future research will concentrate on various techniques to deep learning (DL) to detecting attacks on MANET networks by selecting the decision threshold that explains the techniques' procedures, such as reinforcement neural network (RNN) methods, to provide high learning performance. Including quality of service (QoS) and preventing attacks to network more reliably, reduced memory requirements, and enhanced Scalability of MANET environments. In addition to improving the quality of service (QoS). In addition, we will focus on utilizing the data, but with more reliability and massive size, or what is known as (Big-Data). MANET is capable of simulating and experiment with a greater variety of network topologies, as well as using other methods. Using artificial intelligence (AI) and deep learning (DL) that can detect attacks with greater accuracy and less time enables us to maintain MANET networks with a level of security that surpasses what has been achieved. By further iteration of the iterating process, deep learning algorithms may, in the future, aid in lowering packet loss. Future benefits will also accrue to the network's assumption, which is being bolstered on purpose with the use of 5G technology. The mobile node, then its highly constrained utilization in systems that diverse data collecting, are supported by the 5G communication platform.

REFERENCES

- [1] F. H. Shajin and P. Rajesh, “Trusted secure geographic routing protocol: outsider attack detection in mobile ad hoc networks by adopting trusted secure geographic routing protocol,” *International Journal of Pervasive Computing and Communications*, UK, Emerald Publishing, vol. 18, no. 5, pp. 603–621, Dec. 2020.
- [2] A. Ramasamy Rajeswari, “A Mobile Ad Hoc Network Routing Protocols: A Comparative Study,” in *Recent Trends in Communication Networks*, UK, Intech Open, pp. 1-24, 2020.
- [3] S. S. Jamaesha and S. Bhavani, “A secure and efficient cluster-based location aware routing protocol in MANET,” *Cluster Computing*, USA, Springer, vol. 22, no. S2, pp. 4179–4186, Jan. 2018.
- [4] M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, “A survey of network-based intrusion detection data sets,” *Computers & Security*, Amsterdam, USA, Elsevier, vol. 86, pp. 147–167, Sep. 2019.
- [5] S. Sarkar, G. Choudhary, S. K. Shandilya, A. Hussain, and H. Kim, “Security of Zero Trust Networks in Cloud Computing: A Comparative Review,” *Sustainability*, Basel, Switzerland, MDPI, vol. 14, no. 18, p. 11213, Sep. 2022.
- [6] N. Kunhare, R. Tiwari, and J. Dhar, “Particle swarm optimization and feature selection for intrusion detection system,” *Sādhanā*, USA, Springer, vol. 45, no. 1, May 2020.
- [7] Z. A. Abbood, D. C. Atilla, C. Aydin, and M. S. Mahmoud, “A Survey on Intrusion Detection System in Ad Hoc Networks Based on Machine Learning,” *2021 International Conference of Modern Trends in Information and Communication Technology Industry (MTICTI)*, Sana'a, Yemen, IEEE, pp. 1-8.

- [8] M. Zaid and P. Agarwal, "Intelligent Intrusion Detection System Optimized using Nature-Inspired Algorithms," In *2022 1st International Conference on Informatics (ICI)*, Noida, India, IEEE, pp. 80-85.
- [9] Z. S. Goodwill, "An intrusion detection system for sdn-based tactical networks: a machine learning approach," Ph.D. dissertation, Dept of Computer Science ,Zululand Univ, 2020.
- [10] P. Kumar, G. P. Gupta, and R. Tripathi, "A distributed ensemble design based intrusion detection system using fog computing to protect the internet of things networks," *Journal of Ambient Intelligence and Humanized Computing*, Switzerland, Springer, vol. 12, no. 10, pp. 9555–9572, Nov. 2020
- [11] M. Zhou, L. Han, H. Lu, and C. Fu, "Distributed collaborative intrusion detection system for vehicular Ad Hoc networks based on invariant," *Computer Networks*, Amsterdam, USA, Elsevier, vol. 172, p. 107174, May 2020.
- [12] M. Özalp, C. Karakuzu, A.Zengin, "Distributed Intrusion Detection Systems: A Review," *International Symposium on Innovative Technologies in Eng. and Science*, SanliUrfa, Turkey, pp. 22-24, Nov, 2019.
- [13] S.-W. Lee et al., "Towards secure intrusion detection systems using deep learning techniques: Comprehensive analysis and review," *Journal of Network and Computer Applications*, Amsterdam, USA, Elsevier, vol. 187, p. 103111, Aug. 2021.
- [14] M. Husák and J. Kašpar, "AIDA framework: real-time correlation and prediction of intrusion detection alerts," In *Proceedings of the 14th International Conference on Availability, Reliability and Security*, Canterbury, CA, UK, ACM Library, pp. 1-8, Aug. 2019.
- [15] D. A. Chekired, L. Khoukhi, and H. T. Mouftah, "Fog-Based Distributed Intrusion Detection System Against False Metering Attacks in Smart Grid," In *2019 International Conference on Communications (ICC)*, Shanghai, China, IEEE, pp. 1-6.

- [16] A. A. Ojugo and A. O. Eboka, "Mitigating Technical Challenges via Redesigning Campus Network for Greater Efficiency, Scalability and Robustness: A Logical View," *International Journal of Modern Education and Computer Science*, HK, MECS Publisher, vol. 12, no. 6, pp. 29–45, Dec. 2020.
- [17] H. Fatemidokht, M. K. Rafsanjani, B. B. Gupta, and C.-H. Hsu, "Efficient and Secure Routing Protocol Based on Artificial Intelligence Algorithms With UAV-Assisted for Vehicular Ad Hoc Networks in Intelligent Transportation Systems," *IEEE Transactions on Intelligent Transportation Systems*, IEEE, vol. 22, no. 7, pp. 4757–4769, Jul. 2021.
- [18] N. K. Gupta, R. S. Yadav, and R. K. Nagaria, "3D geographical routing protocols in wireless ad hoc and sensor networks: an overview," *Wireless Networks*, Springer, vol. 26, no. 4, pp. 2549–2566, Apr. 2019.
- [19] M. Elhoseny and K. Shankar, "Reliable Data Transmission Model for Mobile Ad Hoc Network Using Signercryption Technique," *IEEE Transactions on Reliability*, IEEE, vol. 69, no. 3, pp. 1077–1086, Sep. 2020.
- [20] N. Ravi and G. Ramachandran, "A robust intrusion detection system using machine learning techniques for MANET," *International Journal of Knowledge-based and Intelligent Engineering Systems*, IOS Press, vol. 24, no. 3, pp. 253–260, Sep. 2020.
- [21] D. A. Mukeshbhai, and R. H. Jhaveri. "Secure routing in mobile Ad hoc networks: A predictive approach." *International Journal of Information Technology* 11, Springer, no. 2, pp. 345-356, 2019.
- [22] R. M. Hadi, S. H. Abdullah, and W. M. S. Abedi, "Proposed neural intrusion detection system to detect denial of service attacks in MANETs," *Periodicals of Engineering and Natural Sciences (PEN)*, International University of Sarajevo, vol. 10, no. 3, p. 70, May 2022.
- [23] K. R. C. Boni, L. Xu, Z. Chen, and T. D. Baddoo, "A Security Concept Based on Scaler Distribution of a Novel Intrusion Detection Device for Wireless Sensor

- Networks in a Smart Environment,” *Sensors*, MDPI, vol. 20, no. 17, p. 4717, Aug. 2020.
- [24] T. Xie, “A feature selection algorithm combining information gain and multi-objective genetic search for intrusion detection system,” In *MATEC Web of Conferences*, EDP Sciences, vol. 336, p. 08008, 2021.
 - [25] S. Deshmukh-Bhosale and S. S. Sonavane, “A Real-Time Intrusion Detection System for Wormhole Attack in the RPL based Internet of Things,” *Procedia Manufacturing*, Elsevier, vol. 32, pp. 840–847, 2019.
 - [26] N. Koroniotis, N. Moustafa, and E. Sitnikova, “Forensics and Deep Learning Mechanisms for Botnets in Internet of Things: A Survey of Challenges and Solutions,” *IEEE Access*, vol. 7, pp. 61764–61785, 2019.
 - [27] E. E.-D. Hemdan and D. H. Manjaiah, “Internet of Nano-Things Forensics: Performing Digital Forensics in Nanoscale Systems,” In *Internet of Things (IoT)*, CRC Press, pp. 119–138, Oct. 2017.
 - [28] S. Nosratabadi, A. Mosavi, R. Keivani, S. F. Ardabili, and F. Aram, “State of the Art Survey of Deep Learning and Machine Learning Models for Smart Cities and Urban Sustainability,” *Springer International Publishing*, Springer, Oct. 2020.
 - [29] A. Alghazali and Z. Hanoosh, “Using a Hybrid Algorithm with Intrusion Detection System based on Hierarchical Deep Learning for Smart Meter Communication Network,” *Webology international journal*, Webology, vol. 19, no. 1, pp. 3850–3865, Jan. 2022.
 - [30] A. Aldweesh, A. Derhab, and A. Z. Emam, “Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues,” *Knowledge-Based Systems*, Elsevier, vol. 189, p. 105124, Feb. 2020.
 - [31] B. M. Esiefarienrhe, T. Phakathi, and F. Lugayizi, “Node-Based QoS-Aware Security Framework for Sinkhole Attacks in Mobile Ad-Hoc Networks,” *Telecom*, MDPI, vol. 3, no. 3, pp. 407–432, Jun. 2022.

- [32] M. Alloghani, D. Al-Jumeily, J. Mustafina, A. Hussain, and A. J. Aljaaf, "A Systematic Review on Supervised and Unsupervised Machine Learning Algorithms for Data Science," *Supervised and Unsupervised Learning for Data Science*, Springer, pp. 3–21, Sep. 2019.
- [33] M. Panthi, "Anomaly Detection in Smart Grids using Machine Learning Techniques," *2020 First International Conference on Power, Control and Computing Technologies (ICPC2T)*, IEEE, pp. 220–222.
- [34] N. Moustafa, J. Hu, and J. Slay, "A holistic review of Network Anomaly Detection Systems: A comprehensive survey," *Journal of Network and Computer Applications*, Elsevier, vol. 128, pp. 33–55, Feb. 2019.
- [35] T. Su, H. Sun, J. Zhu, S. Wang, and Y. Li, "BAT: Deep Learning Methods on Network Intrusion Detection Using NSL-KDD Dataset," *IEEE Access*, IEEE, vol. 8, pp. 29575–29585, 2020.
- [36] S. Haider et al., "A Deep CNN Ensemble Framework for Efficient DDoS Attack Detection in Software Defined Networks," *IEEE Access*, IEEE, vol. 8, pp. 53972–53983, 2020.
- [37] S. Haider et al., "A Deep CNN Ensemble Framework for Efficient DDoS Attack Detection in Software Defined Networks," *IEEE Access*, IEEE, vol. 8, pp. 53972–53983, 2020.
- [38] I. H. Sarker, "Machine Learning: Algorithms, Real-World Applications and Research Directions," *SN computer science journal*, Springer, Mar. 2021.
- [39] S. Laqtib, K. E. Yassini, and M. L. Hasnaoui, "A deep learning methods for intrusion detection systems based machine learning in MANET," In *Proceedings of the 4th International Conference on Smart City Applications*, ACM Library, pp. 1–8, Oct. 2019.
- [40] E. Jaw and X. Wang, "Feature Selection and Ensemble-Based Intrusion Detection System: An Efficient and Comprehensive Approach," *Symmetry*, Multidisciplinary Digital Publishing Institute, vol. 13, no. 10, p. 1764, Sep. 2021.

- [41] T. T. Khoei, G. Aissou, W. C. Hu, and N. Kaabouch, "Ensemble Learning Methods for Anomaly Intrusion Detection System in Smart Grid," *2021 International Conference on Electro Information Technology (EIT)*, Michigan, USA, IEEE, pp. 129-135.
- [42] S. Parampottupadam and A.-N. Moldovann, "Cloud-based Real-time Network Intrusion Detection Using Deep Learning," *International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, United States, IEEE, Jun. 2018,
- [43] R.-H. Hwang, M.-C. Peng, V.-L. Nguyen, and Y.-L. Chang, "An LSTM-Based Deep Learning Approach for Classifying Malicious Traffic at the Packet Level," *Applied Sciences*, MDPI, vol. 9, no. 16, p. 3414, Aug. 2019.
- [44] S. Swagatika Jena and S. B. B. Priyadarshini, "A Comprehensive Study on Metaheuristics, Big Data and Deep Neural Network Strategies," *Artificial Intelligence and Cybersecurity*, CRC Press, pp. 175–194, Nov. 2021.
- [45] Y. Jia, M. Wang, and Y. Wang, "Network intrusion detection algorithm based on deep neural network," *IET Information Security*, Wiley Online Library, vol. 13, no. 1, pp. 48–53, Jan. 2019.
- [46] P. Rajesh Kanna and P. Santhi, "Unified Deep Learning approach for Efficient Intrusion Detection System using Integrated Spatial–Temporal Features," *Knowledge-Based Systems*, Elsevier, vol. 226, p. 107132, Aug. 2021.
- [47] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," *Journal of Information Security and Applications*, Elsevier, vol. 50, p. 102419, Feb. 2020.
- [48] G. Kocher and G. Kumar, "Analysis of Machine Learning Algorithms with Feature Selection for Intrusion Detection Using UNSW-NB15 Dataset," *SSRN Electronic Journal*, Available at SSRN 3784406, 2021.

- [49] G. D'Angelo and F. Palmieri, "Network traffic classification using deep convolutional recurrent autoencoder neural networks for spatial-temporal features extraction," *Journal of Network and Computer Applications*, Elsevier, vol. 173, p. 102890, Jan. 2021.
- [50] S. Samizade, C. Shen, C. Si, and X. Guan, "Passive browser identification with multi-scale Convolutional Neural Networks," *Neurocomputing*, Elsevier, vol. 378, pp. 238–247, Feb. 2020.
- [51] S. Mishra, R. Sagban, A. Yakoob, and N. Gandhi, "Swarm intelligence in anomaly detection systems: an overview," *International Journal of Computers and Applications*, Taylor & Francis, vol. 43, no. 2, pp. 109–118, Sep. 2018.
- [52] S. B. and M. K., "Firefly algorithm based feature selection for network intrusion detection," *Computers & Security*, Elsevier, vol. 81, pp. 148–155, Mar. 2019.
- [53] T. A. Alamiedy, M. Anbar, Z. N. M. Alqattan, and Q. M. Alzubi, "Anomaly-based intrusion detection system using multi-objective grey wolf optimisation algorithm," *Journal of Ambient Intelligence and Humanized Computing*, Springer, vol. 11, no. 9, pp. 3735–3756, Nov. 2019.
- [54] Y. Yang, H.-Q. Xu, L. Gao, Y.-B. Yuan, K. McLaughlin, and S. Sezer, "Multidimensional Intrusion Detection System for IEC 61850-Based SCADA Networks," *IEEE Transactions on Power Delivery*, IEEE, vol. 32, no. 2, pp. 1068–1078, Apr. 2017.
- [55] C. A. de Souza, C. B. Westphall, and R. B. Machado, "Two-step ensemble approach for intrusion detection and identification in IoT and fog computing environments," *Computers & Electrical Engineering*, Elsevier, vol. 98, p. 107694, Mar. 2022.
- [56] R. Alghamdi and M. Bellaiche, "Evaluation and Selection Models for Ensemble Intrusion Detection Systems in IoT," *IoT*, MDPI, vol. 3, no. 2, pp. 285–314, Apr. 2022.

- [57] S.-Y. Kuo, F.-H. Tseng, and Y.-H. Chou, "Metaverse intrusion detection of wormhole attacks based on a novel statistical mechanism," *Future Generation Computer Systems*, Elsevier, vol. 143, pp. 179–190, Jun. 2023.
- [58] Y. Liao and V. R. Vemuri, "Use of K-Nearest Neighbor classifier for intrusion detection," *Computers & Security*, Elsevier, vol. 21, no. 5, pp. 439–448, Oct. 2002.
- [59] Z. Chen, N. Lv, P. Liu, Y. Fang, K. Chen, and W. Pan, "Intrusion Detection for Wireless Edge Networks Based on Federated Learning," *IEEE Access*, IEEE, vol. 8, pp. 217463–217472, 2020.
- [60] M. Islabudeen and M. K. Kavitha Devi, "A Smart Approach for Intrusion Detection and Prevention System in Mobile Ad Hoc Networks Against Security Attacks," *Wireless Personal Communications*, Springer, vol. 112, no. 1, pp. 193–224, Jan. 2020.
- [61] A. A. Korba, M. Nafaa, and S. Ghanemi, "Hybrid Intrusion Detection Framework for Ad Hoc Networks," *Securing the Internet of Things*, IGI Global, pp. 1312–1346, Sep. 2019.
- [62] R. Meddeb, F. Jemili, B. Triki, and O. Korbaa, "A Deep Learning based Intrusion Detection Approach for MANET," *Serbian Journal of Electrical Engineering*, PMC International, vol. 18, pp. 397–426, Aug. 2022.
- [63] M. V. Rajesh, "Intensive analysis of intrusion detection methodology over Mobile Adhoc Network using machine learning strategies," *Journal of Materials Today: Proceedings*, Elsevier, vol. 51, pp. 156–160, 2022.
- [64] W. Osamy, A. M. Khedr, A. Salim, A. I. AlAli, and A. A. El-Sawy, "Recent Studies Utilizing Artificial Intelligence Techniques for Solving Data Collection, Aggregation and Dissemination Challenges in Wireless Sensor Networks: A Review," *Electronics*, MDPI, vol. 11, no. 3, p. 313, Jan. 2022.
- [65] M. Prasad, S. Tripathi, and K. Dahal, "An intelligent intrusion detection and performance reliability evaluation mechanism in mobile ad-hoc networks,"

Engineering Applications of Artificial Intelligence, Elsevier, vol. 119, p. 105760, Mar. 2023.

- [66] R. T. Adek and M. Ula, “A Survey on The Accuracy of Machine Learning Techniques for Intrusion and Anomaly Detection on Public Data Sets,” *2020 International Conference on Data Science, Artificial Intelligence, and Business Analytics (DATABIA)*, IEEE, pp. 19-27.
- [67] Y. Otoum and A. Nayak, “AS-IDS: Anomaly and Signature Based IDS for the Internet of Things,” *Journal of Network and Systems Management*, Springer, vol. 29, no. 3, Mar. 2021.
- [68] V. K. Quy, V. H. Nam, D. M. Linh, N. T. Ban, and N. D. Han, “A Survey of QoS-aware Routing Protocols for the MANET-WSN Convergence Scenarios in IoT Networks,” *Wireless Personal Communications*, Springer, vol. 120, no. 1, pp. 49–62, Apr. 2021.
- [69] R. S. Krishnan et al., “Modified zone based intrusion detection system for security enhancement in mobile ad hoc networks,” *Wireless Networks*, Springer, vol. 26, no. 2, pp. 1275–1289, Sep. 2019.
- [70] C. Pu and P. Zhu, “Mitigating Routing Misbehavior in the Internet of Drones Environment,” In *Conf. 2022 95th Vehicular Technology Conference: (VTC2022-Spring)*, Helsinki, Finland, IEEE, pp. 1-6.
- [71] A. Ahmim, L. Maglaras, M. A. Ferrag, M. Derdour, and H. Janicke, “A Novel Hierarchical Intrusion Detection System Based on Decision Tree and Rules-Based Models,” In *Conf. 2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, Santorini Island, Greece, IEEE, pp. 228-233.
- [72] X. Wang, D. Zeng, H. Dai, and Y. Zhu, “Making the right business decision: Forecasting the binary NPD strategy in Chinese automotive industry with machine learning methods,” *Technological Forecasting and Social Change*, Elsevier, vol. 155, p. 120032, Jun. 2020.

- [73] S. Gupta and H. N. Jha, "Performance Analysis of MANET Under Grayhole Attack Using AODV Protocol," *Architectural Wireless Networks Solutions and Security Issues (2021)*, Springer, pp. 267–281.
- [74] K. Chawla, J. Gill, and M. Singh, "Comprehensive Survey of IDS Techniques in Mobile Ad Hoc Network (MANET)," *Algorithms for Intelligent Systems*, Springer, pp. 435–449, 2021.
- [75] A. Mohsin, S. Aurangzeb, M. Aleem, and M. T. Khan, "On the Performance and Scalability of Simulators for Improving Security and Safety of Smart Cities," *In Conf. 2022 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*, Stuttgart, Germany, IEEE, pp. 1-8.
- [76] S. O. Johnsen, "Risk Safety and Security in the Ecosystem of Smart Cities," *Risk Assessment*, Intech, Feb. 2018.
- [77] <https://www.kaggle.com/datasets/hassan06/nslkdd> (accessed Jan. 09, 2023)
- [78] C. Yin, Y. Zhu, J. Fei, and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," *IEEE Access*, IEEE, vol. 5, pp. 21954–21961, 2017.
- [79] S. Laqtib, K. El Yassini, and M. L. Hasnaoui, "A technical review and comparative analysis of machine learning techniques for intrusion detection systems in MANET," *International Journal of Electrical and Computer Engineering (IJECE)*, Institute of Advanced Engineering and Science publisher, vol. 10, no. 3, p. 2701, Jun. 2020.
- [80] S. Laqtib, K. E. Yassini, and M. L. Hasnaoui, "A deep learning methods for intrusion detection systems based machine learning in MANET," In *Proceedings of the 4th International Conference on Smart City Applications*, New York, NY, USA, ACM, pp. 1-8, Oct. 2019.

