ISOLATION OF MALICIOUS NODES BY COMPUTATION OF ENERGY LEVEL IN A TRUST BASED MECHANISM FOR A SECURE IOT NETWORK

Thesis Submitted for the Award of the Degree of

DOCTOR OF PHILOSOPHY

in

Computer Science and Engineering

By

Zakiya Manzoor

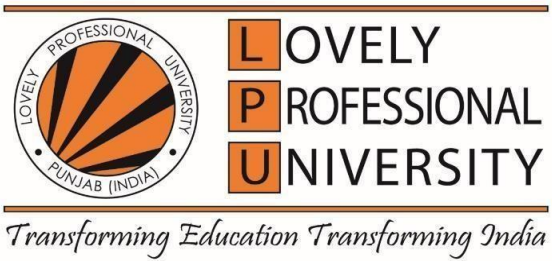
Registration Number : 41800963

Supervised By

Dr. Harjit Singh (14952)

Computer Science and Engineering (Associate Professor)

Lovely Professional University



LOVELY PROFESSIONAL UNIVERSITY, PUNJAB

2025

CANDIDATE'S DECLARATION

I hereby declare that the thesis entitled, **ISOLATION OF MALICIOUS NODES BY COMPUTATION OF ENERGY LEVEL IN A TRUST BASED MECHANISM FOR A SECURE IOT NETWORK** submitted for the Degree of Doctor of Philosophy in Computer Science and Engineering is the result of my original and independent research work carried out under the guidance of Supervisor Dr. Rajni, Associate Professor, School of Computer Science and Engineering, Lovely Professional University, Punjab and Co-Supervisor. This work has not been submitted for the award of any degree, diploma, associateship, fellowship of any University or Institution.

Date: 12-03-2025

Zakiya Manzoor

Registration No: 41800963

CERTIFICATE

This is to certify that the thesis entitled **"ISOLATION OF MALICIOUS NODES BY COMPUTATION OF ENERGY LEVEL IN A TRUST BASED MECHANISM FOR A SECURE IOT NETWORK"** submitted by Zakiya Manzoor for the award of degree of Doctor of Philosophy in Computer Science and Engineering, Lovely Professional University, is entirely based on the work carried out under my supervision and guidance. The work reported, embodies the original work of the candidate and has not been submitted to any other university or institution for the award of any degree or diploma, according to the best of my knowledge.

Signature of Supervisor

Name: Dr. Harjit Singh

Date: 12-03-2025

ABSTRACT

New network services and Internet users are growing. Threats to network security have grown in severity in tandem with the expansion of the internet. Attacks reveal & exploit numerous security holes. According to recent reports, both the frequency & cost of Internet security breaches are steadily increasing. The utilization of network traffic has become a common tactic in recent network attacks. Threatening networks or hosts without actually compromising them is possible when an attacker uses this tactic. A few instances include assaults on well-known websites like Amazon, Yahoo!, & eBay. Even inexperienced hackers can find and utilize one of the many available DoS/DDoS programs.

Effective network management relies on the timely and precise identification of traffic anomalies. Time of failure in a network is typically hard to pinpoint. The effects of a successful DoS/DDoS attack are noticeable rapidly, and the invader is sometimes difficult to track down. Additionally, the attacks have an effect on network performance because of the bandwidth they need. By monitoring & evaluating system events, security software seeks to identify user actions that violate the system's intended purpose. Given the similarity in representation between "regular" & "minority" traffic data types, the classifier has difficulty learning to distinguish between the two in situations where there is a large quantity of both types of network traffic. As may be observed in its linked samples, the bulk of the training set contains duplicated noise information.

The majority class is compressed to help the classifier grasp the minority class's distribution, since it is significantly larger than the minority class. While the minority group's discrete traits remain the same, their continuous traits show clear differences. Zooming in on the continuous characteristics of the minority class yields data that is more true to the true distribution. Our recommendation is the DSSTE algorithm to fix this discrepancy. For the purpose of identifying malevolent nodes, the threshold method has been suggested in previous studies. Detecting malicious nodes using the threshold technique is not as accurate as other methods.

In this study, we propose using the Received Signal Strength Indicator (RSSI) to pinpoint the exact location of both malicious & IoT nodes. Trust-based approach is providing as a method for verifying malicious nodes. It is planned to use the proposed method, and its effectiveness in reducing throughput, delay, & packet loss is anticipated.

ACKNOWLEDGEMENT

I would like to present my deepest gratitude to **Dr. Harjit Singh** for their guidance, advice, understanding and supervision throughout the development of this thesis and study. Despite their busy schedule they have been available at every step, devoting time and energy and the much-needed counsel and advice. This enabled me to sail through the tough times and complete this enormous task.

I would like to thank to the **research project committee members** for their valuable comments and discussions. A special thanks to the management of **Lovely Professional University** for their support in academic concerns and letting me involve in research study. The doctoral program of LPU has made it possible for me to pursue my dream of research and upgrade my knowledge.

My sincere feeling of gratefulness also goes to my parents and family members who always motivated me in all the endeavors of my life including this research work in LPU. I would like to thank each person who has directly and indirectly helped and motivated me in this journey.

Zakiya Manzoor

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATION	DESCRIPTION
IOT	INTERNET OF THINGS
ML	MACHINE LEARNING
AI	ARTIFICIAL INTELLIGENCE
NLP	NATURAL LANGUAGE PROCESSING
NN	NEURAL NETWORK
RNN	RECURRENT NEURAL NETWORKS
CNN	CONVOLUTION NEURAL NETWORKS
DT	DECISION TREE
RF	RANDOM FOREST
SVM	SUPPORT VECTOR MACHINE
LR	LOGISTIC REGRESSION
KNN	K-NEAREST NEIGHBOR
NB	NAÏVE BAYES
HDFS	HADOOP DISTRIBUTED FILESYSTEM
HIVE	HADOOP SCRIPTING LANGUAGE
DM	DATA MINING
DL	DEEP LEARNING
SA	SENTIMENT ANALYSIS
DCNN	DEEP CONVOLUTION NEURALNETWORK

MLP	MULTILEVEL PERCEPTRON
PCA	PRINCIPAL COMPONENT ANALYSIS
DOS	DENIAL-OF-SERVICE
WSN	WIRELESS SENSOR NETWORKS
LLN	LOW POWER AND LOSSY NETWORKS
DODAG	DESTINATION ORIENTED DIRECTED ACYCLIC GRAPHS
R2U	REMOTE TO USER ATTACKS
U2R	USER TO ROOT ATTACKS
IDS	INTRUSION DETECTION SYSTEMS
RDD	RESILIENT DISTRIBUTED DATASET
HIDS	HOST BASED INTRUSION DETECTION SYSTEM
NIDS	NETWORK BASED NIDS
DSSTE	DIFFICULT SET SAMPLING TECHNIQUE
LSTM	LONG SHORT TERM MEMORY
AE	AUTOENCODER

Chapter – 1

INTRODUCTION

1.1 INTERNET OF THINGS

IoT schemes have drawn a lot of interest as a way to collect sensory data and create intelligent services & applications. IoT is described as the method of connecting any device to the internet using embedded software & sensors for data collection, sharing, and communication. The IoT makes the world much more affordable & provides an almost infinite number of opportunities & connections at home, at work, and while playing. A sort of natural communication is formed between people & computers, software, and hardware as a result of the connecting of sensors, devices, & people through the IoT. In the same way that internet networks & computer displays have improved the information world thanks to AI & ML [1], these discussions make possible devices that can forecast, react, respond, and alter the physical world.

This potential is known as the Connected Life by the GSMA. Experts anticipate rapid expansion of the IoT in the next years, ushering in a plethora of new services within this convergence that will enhance both the lives of consumers & productivity of businesses.

The IoT can provide clients with solutions that improve their daily lives in terms of energy efficiency & number of other areas [2]. This technology has given businesses useful options for enhancing decision-making & output in the manufacturing, retail, agricultural, and other sectors.

Figure 1.1 represents the entire IoT system scenario in a diagram form, representing a variety of physical devices in which interconnectivity has been recognized between home appliances, buildings, personal computers, & cars.

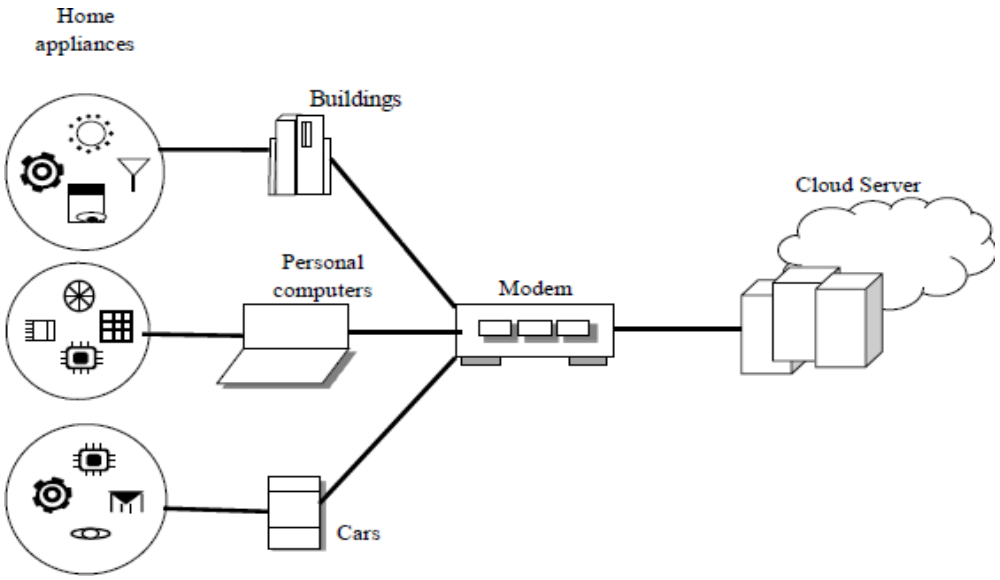


Figure 1.1: Representation of IoT Model

All the necessary components, including sensors, actuators, electronics, software, & network connectivity, are present in the environment, allowing for practical communication between all the linked devices. Internet Protocol (IP) addresses and unique identifiers are assigned to every node in an IoT network. Its M2M nature has been confirmed [3]. In order to carry out the owner's specified tasks, the items linked to this network communicate with one another and share data.

1.1.1 IoT Applications

Nearly all IoT applications have adopted the label smart, including Smart Home, or connected Health. The recent primary applications have included:

- **Smart home:** The phrase "Smart Home" refers to the connectivity within homes. Smoke detectors, light bulbs, devices, entertaining systems, windows, door locks, and etc. It includes some well-known brands including Nest, Apple, & Philips.

- **Wearables**: Whether it was the Jawbone Up, Fitbit Flex, or Apple Smartwatch, it helped to create a sizable portion of the client base for IoT applications.

- **Smart City:** This category contains a inclusive range of cases, including traffic control, water distribution, waste management, city security, & ecological

monitoring. It helped to lessen the actual suffering of individuals who now live in cities. It is useful for making cities safer, reducing noise & pollution, and resolving traffic congestion difficulties.

- **Smart grids**: They allow for the automated distribution of information about the habits of electrical suppliers & users, improving the economics and efficiency of electricity.
- **Industrial internet:** Several commercial analyses have concluded that the industrialized internet represents the most promising IoT concept.
- Amongst other things, the applications include networked industrial equipment or smart factories.
- **Connected Health (Digital Health/Telehealth/Telemedicine):** Smart medical equipment & connected health care system have shown to be effective for both businesses and individuals in common. [4].
- **Smart agricultural**: The inclosing becomes a noteworthy IoT use case because to the separation of agricultural processes & substantial amount of easily trackable cattle.

1.1.2 Security in IOT

In the perspective of information technology, security concerns have long been present. Regularly, new and distinct security concerns arise as a result of characteristics of IoT deployments with many functionalities. Consequently, we must resolve these concerns & prioritize the preservation of security in IoT goods & services. It is essential that people using IoT devices have faith in the safety of those devices and the data services connected to them. Because technology is now an integral part of everyone's daily lives, this is of the utmost importance.

IoT services & devices that are not secured could be used as entry points for cyberattacks. By leaving data streams poorly safeguarded and exposing user data to theft, the attackers can take advantage of this situation. Due to the interconnected nature of Iot devices, any insecure device associated with the Internet could

negotiation the general security & flexibility of the Internet. Additional considerations like as the introduction of uniform IoT devices, the capacity of certain devices to establish connections with other devices on the fly, and the possibility of exposing these devices to hazardous conditions amplify this difficulty [5]. Manufacturers & consumers of IoT devices & systems have an ethical need to take reasonable precautions to prevent damage to themselves & Internet. Effective and appropriate solutions to the security difficulties of the IoT that are well-suited to the volume & complexity of these issues will need a collaborative approach to security.

Security Attacks in IoT

The following list includes some IoT security attacks that are frequently launched:

- **DoS attacks:** As a form of Denial of Service, sleep deprivation, power outages, & battery depletion attacks all target computing nodes, but each has its own unique characteristics. Lack of sleep is a form of Dos attack. During this attack, battery-operated nodes may be subjected to a flood of requests. These questions appear to be genuine & have been sent by an intrusive party. IoT nodes are deployed with smaller batteries due to their modest size. Battery-draining intrusions, which can have negative effects like a power outage, are primarily caused by this. Attacks of the third kind happen when an IoT object is unable to complete the tasks that are expected of it. Possible causes of this include lack of sleep, code inoculation, & unnecessary manufacturing mistakes.

- **Physical attack:** It is necessary to use some IoT-related products in dangerous settings, which leaves them open to physical access & potential hardware or firmware attacks. [6]. An adversary with physical access to an item can compromise it in a number of ways: they can alter the operating system, destroy hardware, & steal crucial cryptographic data.

- **Side channel attack:** IoT devices can accidentally leak crucial information because they function regularly. Even if they don't employ a wireless protocol to transmit the data, it can still be utilized.

- **Eavesdropping**: There is a potential that this attack—which typically includes communication protocols—will take place here, particularly for RFID tags. An

eavesdropping attack's main objective is to intercept, read, & manipulate messages so that more research may be done.

- **Tag cloning:** Although this kind of attack is incredibly advantageous to hackers, it could also be harmful to a company's reputation. An attacker can access secure areas & secret information by copying the tags [7].

- **Collision attacks:** The link layer is vulnerable to collision-causing attacks. An adversary could purposefully disrupt communication channels in order to bring about a collision, for example. In this collision, it is important to retransmit packets that were damaged during the collision. Since this form of attack results in multiple collisions and retransmissions, it can quickly drain a target's battery.

1.2 VERSION NUMBER ATTACK IN IOT

IoT is an emerging technology that is rapidly gaining traction within the context of international networks & services. This model broadens the scope of the Internet to encompass tangible objects that may coordinate their actions in the physical environment through communication and interaction. Due to the fast growth of this technology, LLN (Low power & Lossy Networks) have been widely used, especially in home automation systems & WSNs. The RPL routing method is famous for IPv6-based LLN. These restrictions make RPL-based networks susceptible to various security threats [8].

The RPL function, typically employed to ensure a topology devoid of loops and errors, can be exploited by the version number attack. Because a malevolent node altered the topology-associated version number, the routing tree had to be rebuilt from scratch. The control messages from the parents include the version number, however the stated version number cannot be verified according to the protocol that has been set up. The energy supply may decrease, channel availability may be affected, and routing topology loops could occur as a consequence of a forced rebuild. According to certain studies, such attacks significantly affect RPL networks, underscoring the necessity of resolving them.

An IPv6-based distance-vector routing protocol is referred to as the RPL protocol. A specific architecture forms the foundation for the interconnectivity of RPL devices.

This topology creates DODAG by fusing mesh & tree topologies. The root node, also known as the data sink, is the starting point for constructing a DAG. As can be seen in Figure 1.2, a network may utilize a single RPL instance or multiple RPL instances, each of which may host a different number of DODAG graphs. Every occurrence of RPL serves some purpose in an underlying objective function. A set of parameters or constraints may be used to measure the best route using this objective function. This function, for instance, can compute the direct route or minimize energy use. An RPL node can be connected to many different instances at once, but it can also be connected to just one DODAG graph example, like nodes 13 &17 in the figure.

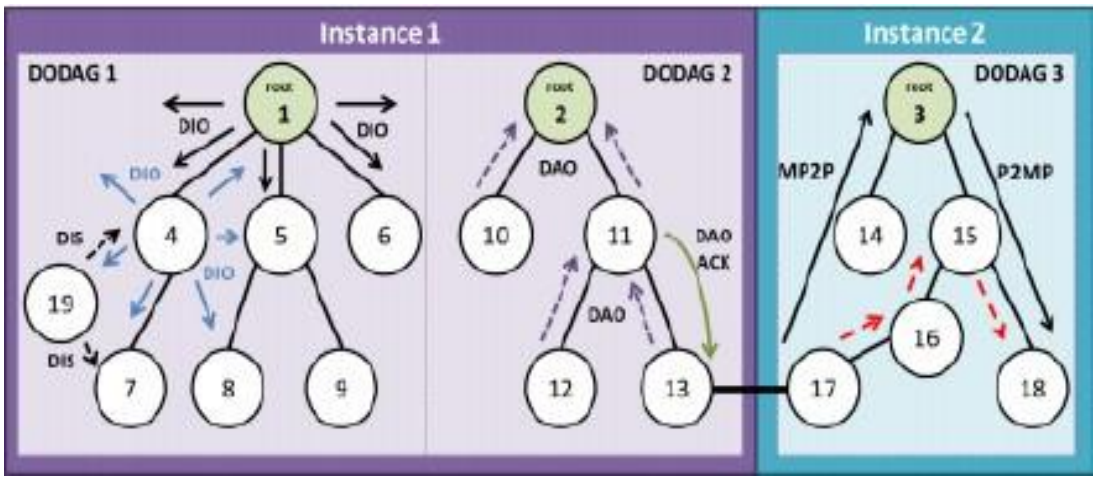


Figure 1.2: An illustration of an RPL network with 3 DODAGs & 2 instances

It is possible for the RPL protocol to do various optimizations, similar to quality-of-service (QoS) optimizations, on these manifold instances. Figure shows that the third DODAG allows RPL packets to be forwarded based on the three traffic patterns. Firstly, there's MP2P, which involves the upward routing of RPL packets from leaves to the root. Using downhill pathways, the following traffic pattern known as P2MP transfers RPL packets from the root to the leaves [9]. The third pattern of P2P traffic is indicated by the red doted arrows. This traffic pattern forwards packets using both upstream & downstream pathways.

Version number metrics indicate a worldwide repair effort in the RPL network. Alterations to this topology-rebuilding metric can only be made by the DODAG root, by the way. A crucial component of every DIO message is the version number. Unimpeded, the version is disseminated along the DODAG graph. Only the root can increase it when the DODAG needs to be rebuilt, which is also called global repair. Inthe new DODAG graph, a node with an older value means it has not been elevated to parent node status. By passing the DIO messages to its neighbors to update the version number, an intrusive party can illegitimately enlarge this area of DIO communications. During this infiltration, the entire DODAG graph undergoes redundant reformation [10]. This indicates that the intrusion may create a number of loops as a result of which different data packets may be lost. Additionally, the repeated duplicate reformations of the graph significantly increase the control message cost, causing network congestion & depletion of the nodes' resources. Compromise nodes cannot impersonate the root or transmit an unauthorized larger Version Number according to a security method called VeRa (Version Number & Rank Authentication). This strategy uses authentication mechanisms that rely on hashing procedures. If so, a node can quickly determine whether the root node or another malicious node has altered the Version Number. In this manner, it is impossible to assume the identification of the DODAG root.

1.3 NETWORKING ATTACKS

In this part, we define the four most common types of network attacks. Every network attack comes into one of these categories.

- **DoS -** DoS, the most destructive sort of attack, prohibits memory resources from serving networking requests, denying customers access to a system.
- **Remote to User Attacks (R2U) -** An attacker conducts this kind of attack by sending unwelcome packets to a computer through the internet in an attempt to obtain access to its resources & determine its weaknesses.
- **User to Root Attacks (U2R)-** U2R are attempts to exploit system flaws in order to attain super user access, such as using xterm or perl, by first logging into the system with a regular user account.

- **Probing -** A vulnerability assessment is a type of attack that examines a computer or networking device for potential security holes. These methods are frequently employed in data mining, such as portsweep, mscan, nmap, saint, etc.

1.4 INTRUSION DETECTION SYSTEMS

Analysis of computer system activity by monitoring and reporting or networked computer system to recognize user behavior that conflicts with the system's intended purpose is known as intrusion detection. As depicted in Figure 1.3, an IDS usually runs behind the firewall and scans network data for patterns that might point to malicious activity. IDSs are employed in any secured network as the second & ultimate line of defense against attacks that get past other barriers.

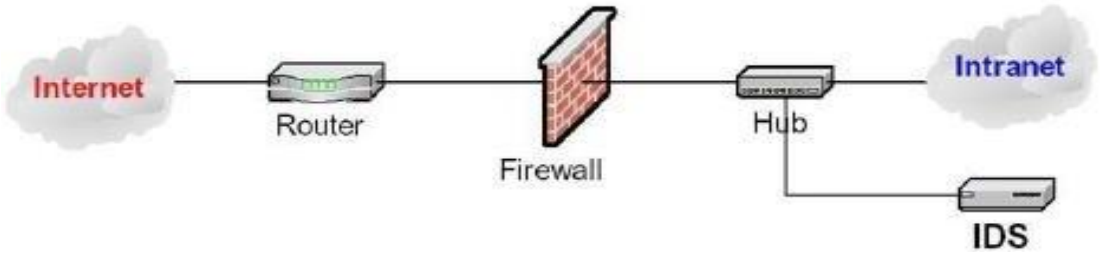


Figure 1.3 : IDS Process

Cryptography, firewalls, & other network security measures aren't made to withstand threats at the network & application layers, such as worms, viruses, Trojan horses, & DoS/DDoS attacks. The fast growth of the Internet and the prevalence of online threats have drawn the attention of security experts to IDS. In the event of a network attack, these technologies will detect it and respond accordingly. This is the set of procedures used to identify potentially malicious actions on computers & networks.

The two most popular approaches to building an intrusion detection system are signature-based & anomaly-based. When scanning for malicious activity, a signature-based IDS uses previously identified patterns to draw conclusions about the source. An anomaly-based IDS, in contrast, monitors network activity for anything out of the ordinary. Instances of abnormal traffic patterns include both the user's destruction of the legal profile established for authorized activity & exceeding of the permitted

standards for the frequency of occurrences in a connection. After a regular traffic profile has been developed in the training phase, anomalies can be detected in the detection phase by applying the profile to the current traffic & looking for deviations from the norm. Recently, a number of techniques for spotting these outliers have been published; broadly speaking, these fall into one of three categories: statistical methods, DM methods, & ML-based methods. In this study, we discover the potential of combining data mining with machine learning.

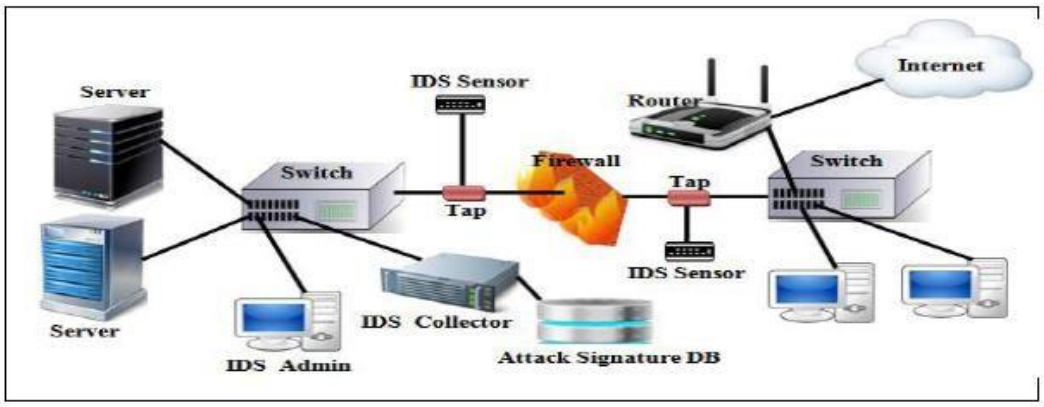


Figure 1.4: Signatures – Based IDS

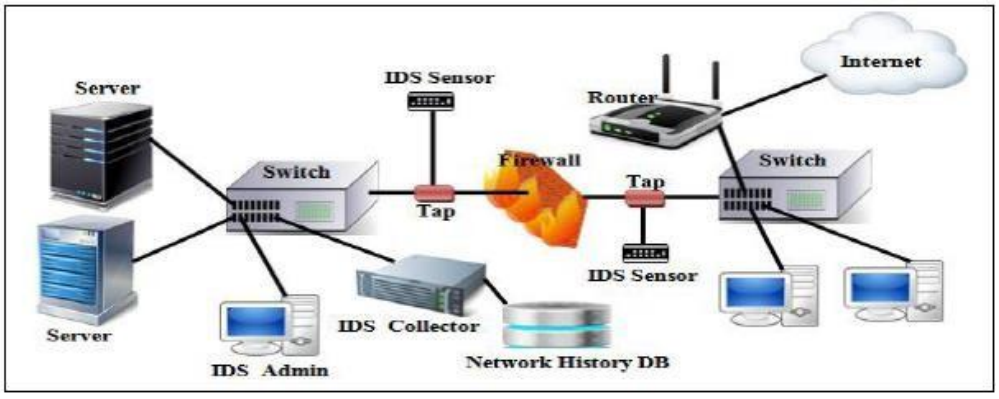


Figure 1.5: Anomaly – Based IDS

1.4.1 Real-Time IDS

In a networking environmental, data is delivered in a variety of ways at very fast speeds. In order to handle the high velocity, volume, & variety of data, a reliable system is needed. These are commonly referred to as "big data." It can be formally structured, loosely structured, or unorganized. The IDS must be effective enough in a

large data environment to process various types of data at high transmission speeds without dropping or losing any crucial flow packets.

Due to the inability of current security systems to identify them, cyberattacks on big data are on the rise at the moment. There are many IDS available for various kinds of network attacks. The majority of them lack the ability to recognize recent, unidentified attacks, while others do not offer an immediate remedy for problems. The distribution,accuracy, speed, and intelligence of intrusion detection studies are on the rise and will incorporate the subsequent techniques.

- ***Distributed Intrusion Detection***-A distributed IDS uses collaborative processing, distributed structure, & various analysis of information, as well as a single architecture of IDS compare with high detection ability. Distributed IDS are typically for heterogeneous systems & large networks.

- ***Intelligent intrusion detection***- It is presently in development & uses techniques as ML, DM, & neural networks. In the use and research of the IDS, it has disseminated a variety of intelligent strategies. Reduced false alarm probability, improved system self-learning, & decreased false alarms are the main goals of the study.

- ***High-Speed Packet Capture Technology***-High-speed packet capture enables network IDS to detect threats more quickly and with less resource usage.

- ***Effective Pattern Matching Algorithm***-Sophisticated models must be stored in the rule base as invasions get more complex & varied. The pattern matching algorithm has to be enhanced & improved as a result.

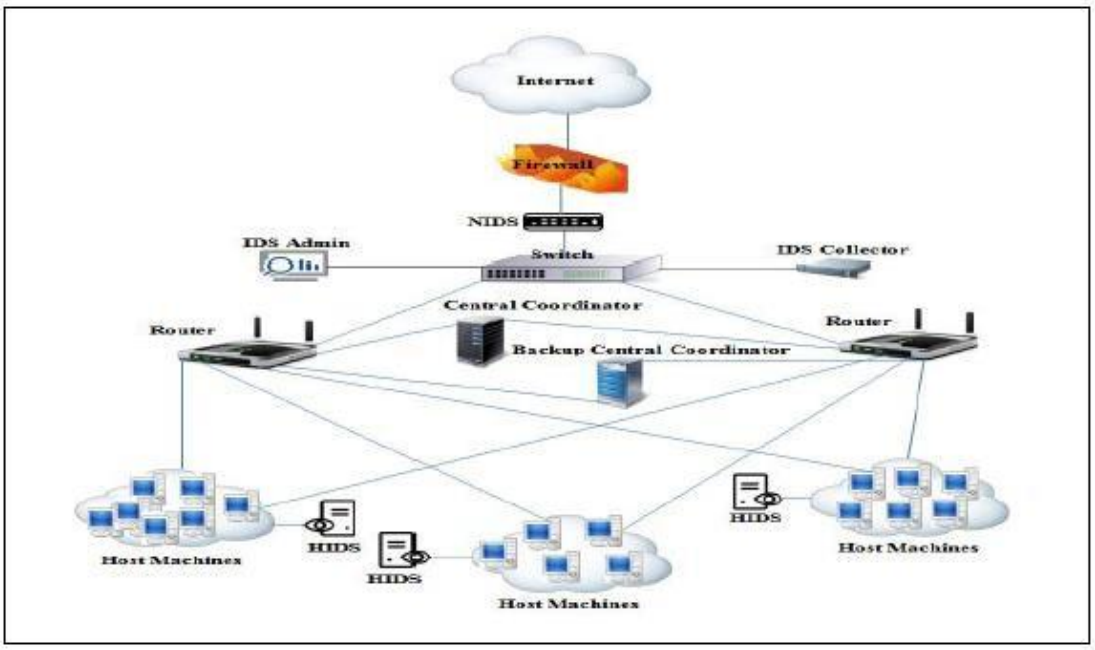


Figure1.6: Collaborative IDS architecture for Big Data

1.4.2 Apache Spark

Fast processing times & lot of storage space are requirements for applications that deal with large amounts of data and fast processing speeds. Regardless, distributed systems like Hadoop can offer scalable distributed storage space. Unfortunately, it can't handle data efficiently because of its built-in processing method for multi-disk I/O operations. We used the Apache Spark framework to fix this.

Try Apache Spark, an open-source clustering technology, for quick data processing. Data transformations include a number of steps, and Spark gives you the ability to temporarily store those steps' output in memory. When compared to Hadoop, Spark's performance for cache/memory & disc data is up to 100 times better, and its processing speed is 10 times faster. One of the key ideas behind Spark is that it offers an abstraction of the Resilient Distributed Dataset (RDD), a group of objects that can be explored concurrently & divided up among the working nodes. R DD form is immutable when stored in read-only mode, which is another important RDD attribute. They can be cached by Spark for computations. Despite the fact that RDDs are immutable, Spark provides a number of functions that can be used to change the RDD from one state to another, including map, reduce, flatmap, filter, take, collect,

& count. Additionally, Spark has helpful built-in libraries such (Spark SQL, MLlib, & Dataframe). These APIs were created by the best spark developers to offer a framework that provides dataset abstraction. On datasets that are quite comparable to those in the database, Spark SQL offers relational operations. The RDD is transformed into a database, which is comparable to a table in the Spark frame, and it operates on that frame (RDBMS). Users can use Spark SQL to interact with data in the Spark database.

1.5 TYPES OF IDS

HDS & NIDS are the two primary categories into which IDS can be divided according to the range of protection (or by location). The sections that follow give explanations of these systems.

1.5.1. HIDS

The contents of the operating system, system files, & applications on one or more host computers are among the data that HIDSs examine [7]. High-Level Intrusion Detection Systems (HIDS) observe system program execution & user behavior in addition to gathering information from computer-internal sources, frequently at the operating system level (various logs, etc.).

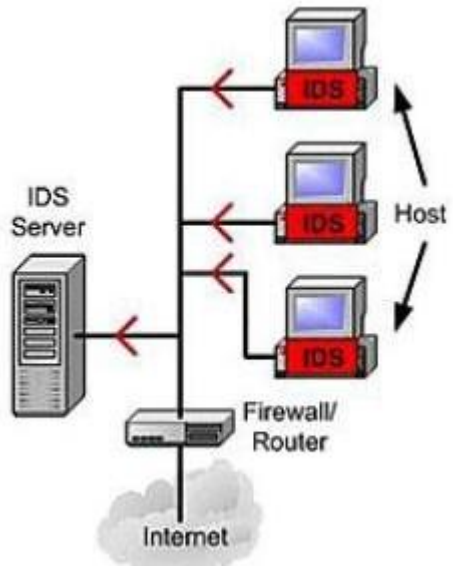


Figure 1.7: Host Based IDS

Advantages of HIDS

Some benefits of host-based IDS versus network-based IDS do occur, despite the fact that it is generally not as effective as NIDS:

- **Descriptive logging** - HIDS may gather a great deal more specific data on precisely what takes place throughout an attack.
- **Better recovery** - Recovery after a successful incident is typically more thorough due to the higher granularity of tracing events in the monitored system.
- **Detection of unknown attacks** – HIDS identifies unknown attacks more frequently than NIDS because they upset the monitored host.
- **Less false positives** - Due to the way HIDS operates, it generates less false warnings than NIDS.

Disadvantages of HIDS

- **Incomprehensible data –** The wide range of networks & operating systems makes it impossible for a single HIDS to offer universally accurate translations for all network applications, operating systems, & file systems. Lacking a corporate key or equivalent method, no IDS can decode encrypted data either.
- **Indirect information -** A system or application's audit record of activity is typically used by HIDS instead of actively monitoring activity. This audit record varies substantially in both quality and quantity among systems & applications, which significantly affects the effectiveness of IDS.
- **Complete coverage:** The system under observation has HIDS installed. Several thousands of workstations may be present on very large networks. This much IDS is both incredibly expensive & very challenging to handle.
- **Outsiders** - Unlike network-based IDS, a HIDS may be able to detect an outside attacker once they have entered the monitored host system. Network security must have allowed the intrusion to reach the host system.

- **Interference with normal host operations:** HIDS places a heavy demand on the host CPU. An IDS audit record may produce unnecessary loads on some systems.

1.5.2 Network Based IDS (NIDS)

It detects intrusions by keeping an eye on network device traffic (e.g. Network Interface Cards, switches & Routers). Its data is primarily collected by generic network streams, like internet packets, that travel through networks. Only NIDS can identify every attack in a LAN or can identify attacks like DOS that HIDS cannot identify.

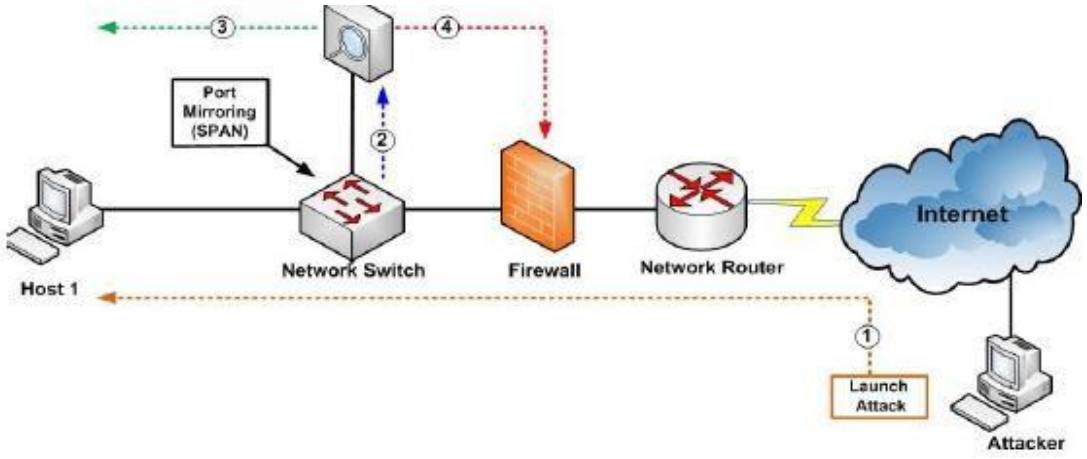


Figure 1.8: Network Based IDS

The following are some of the key reasons why an NIDS is necessary:

1. Increase the infrastructure of an organization's integrity to a higher extent.
2. Capable of following user activity from the attack's input point to its departure point.
3. Document data modification & provide a report.
4. Assist in keeping track of recent attacks on the internet.
5. Alert when a system is being attacked.
6. Analyze any unusual activity patterns.

The vast majority of IDS now in operation suffer from at least two of the problems listed below [9]:

1. Network packets or audit trails provide the information that the IDS uses. An attacker may damage or manipulate data as it travels to the IDS. In addition, the data collected is used by the IDS to extrapolate the system's behavior, which can introduce mistakes or cause incidents to be missed. The fidelity problem is the name for this.

2. Because the IDS's components must always be running, it increases resource consumption in the monitored system even when no intrusions are occurring. Resource utilization is problematic. The IDS's components are separate programs, making them hackable. If an intrusion disables or alters the already running programs on a system, the IDS may be rendered useless or ineffective. This is the issue with dependability.

Advantages of NIDS

- **Ease of deployment -** Because of its passive nature, the supervised environment has few performance or compatibility problems.
- **Cost -** Strategically positioned sensors can be used to monitor a wide organizational environment, as opposed to a HIDS, which required software on each monitored server.
- **Range of detection -** Compared to host-based IDS, network traffic analysis can identify a greater range of hostile activity.
- **Forensics integrity -** The network-based IDS sensors are more resistant to manipulation because they are hosted on a host that is different from the target.
- **Detects all attempts –** In contrast to NIDS, When an attack fails, the monitored host suffers no immediate consequences, hence HIDS only detects successful attacks.

Disadvantages of NIDS

- **Susceptibility to direct attacks**: According to a recent Secure Networks, Inc. analysis of top NIDS systems, NIDS are vulnerable to:

- Spoofing packets to make an IDS think they came from a different source.
- Packet fragmentation attacks, which involve retransmitting sequence numbers to trick the IDS into receiving only the data the hacker wants it to.

- **Indecipherable packets -** Due to network heterogeneity & relative abundance of protocols, network-based IDSs frequently are unable to understand the packets they collect. Encrypted data cannot be decrypted by an IDS without a corporate key.

- **Failure when loaded -** Recent research into the best network-based commercial products has revealed that even those that successfully identify all tested assaults on an empty or moderately used network start to miss at least some attacks when the monitored network is heavily loaded.

- **Failure at wire speed** - NIDS systems typically claim support for networks operating at 10Mbps, although being capable of processing packets at 100Mbps or higher.

- **Lack of comprehensive coverage –** Most sensors are designed to be used on joint access segments and can therefore only monitor traffic on those particular segments. For adequate coverage, the IDS user must select strategic shared-access segments for IDS sensors. Sensors are commonly set up in the demilitarized zone & front of data centers & port facilities. Monitoring remote access points, local area networks, wide area networks, and individual workstations requires a large number of sensors to be strategically deployed. Dial-up connections from a desktop computer or illicit modem use will still be unmonitored.

- **Switched networks -** Unfortunately, switched networks have replaced shared/routed networks as the standard architecture. By abandoning the use of IDS products based on shared-access networks, traffic secrecy is maintained. Because of the dispersed nature of switched networks and the need for several sensors to ensure coverage, the deployment of shared-access IDS is often cost-prohibitive. Alternatives include mirroring selected data, such as those

travelling to certain essential devices, to a sensor for processing or connecting hubs to switches whenever switched traffic needs to be monitored. None of these are quick fixes or the best options.

- **Insiders -** Network-based IDS focuses on finding external attacks rather than trying to find insider abuse & local security policy breaches.

1.6 MACHINE LEARNING APPROACHES

The study of learning strategies that help computer programs adapt their behavior based on prior observations is known as ML, a subfield of AI. Because of their potential to aid users in adjusting to new circumstances and gaining new knowledge. In general, a learning technique transforms outside knowledge into a cutting-edge idea for future application.

Classification & prediction are two categories that can be used to group machine learning systems. The specifics of classification & prediction are briefly described in this section.

1.6.1 Classification

Data mining literature states that classification is a two-stage procedure that necessitates both learning & categorization.

- Learning: The model or classifier produced by applying a classification algorithm to a set of training data can be represented as a set of rules for making such distinctions.

- Classification: The efficacy of the classification rules is using experimental data. Assuming sufficient precision, the rules can be employed to categorise fresh data tuples.

- Due to the prior knowledge of the training tuple's class label, this method is classified as supervised learning. Supervised learning methods include things like SVM, NN, DT, KNN, & GAs. Supervised categorization, a common problem for AI systems, is the primary concern of this thesis.

A classifier's confidence on a given test set is not dependent on the known class labels of each training tuple and an unknown number or set of classes, but rather on the proportion of properly classified tuples. Unsupervised learning, often called clustering, is the opposite of this.

1.6.1.1 Supervised Learning

- **Decision trees -** A subject is led through a series of decisions via DT, with the outcomes of those decisions affecting the subsequent decisions. These kinds of decision-making processes can be represented using trees. The process of categorizing a sampling begins at the root node & works its way out to the correct end leaf node, where each of the groups represented by the nodes indicates a different categorization. Each node is given a specimen's characteristics, and each branch's value corresponds to the result. Examples are C4.5, ID3, & CART (Classification & Regressing Tree).

- **K-Nearest Neighbor -** The simplest and most common nonparametric method for classifying specimens is KNN. To achieve this, it uses the estimated distances between points on the input vectors to assign the unlabeled point to a KNN class. Developing k-NN classifiers relies heavily on the value of K, as different values yield varied outcomes. The accuracy of the forecast will be affected by the time required to classify the neighbors utilized for prediction, which increases as k grows larger. It is known as precedent-based learning & distinct from the concept of preparatory learning. As a outcome, it does not include the model training stage; instead, it just looks through input vector samples and classes prior data. As a result, k-NN "on-line" trains the instances & identifies the precedent's KNN.

- **ANN -** ANN is a type of computing architecture that model themselves after biological nerve systems as human brain. ANN can be fine-tuned for jobs like data classification and pattern identification via training. ANNs are an AI subfield that purposes to pretend the workings of the human brain. Neural networks work by establishing connections between processing components, which are like the computer's neurons, as opposed to the digital paradigm, which

only uses ones & zeroes to accomplish computations. The connections' hierarchies and weights dictate the output. An activation function, a network architecture, and the option to assign weights are the three fundamental components of any ANN. A few examples of popular network types are recurrent networks, multilayer feedforward networks, & single layer feedforward networks.

- **Support Vector Machines (SVM):** To find the best separating hyper-plane, SVM conducts a search in a higher-dimensional feature space into which the input vector has been transformed. The decision boundary, sometimes called the separation hyper-plane, is extremely robust against outliers since support vectors are employed instead of the full training set to create it. The system's major function is for making binary distinctions. In addition, the SVM allows for a penalty factor, which is a user-defined criterion. Users can evaluate a decision boundary's width & number of misclassified samples in relation to one another.

- **Genetic algorithms**: A computer is used to simulate natural selection & evolution using this evolutionary technique. The idea behind this term is derived from "adaptive survival in natural creatures." First, a large number of candidate programmes are generated at random. Each member of a population is evaluated based on some sort of fitness metric. Crossover and Mutation processes create recombination for new population, followed by a huge number of rounds to choose the fittest chromosomes. The method of GA used in the present study is adaptive GA. This tool determines the anticipated generational divide, or the proportion of the population that will be uprooted by newcomers. It makes adjustments to the mutation number & selects the crossover solution depending on the present status of the population. Examining at some of the population's characteristic values, such its average fitness level or its greatest and worst performers, allows one to evaluate the population's condition.

- **Rough Set Approach -** When working with noisy or approximative data, this technique can be used to probe underlying structural links. This means that

normalizing continuous-valued properties is a prerequisite to using them. The foundational principle of RST is to construct equivalence a platform using the supplied training data. All the data tuples in an equivalence class have the same or very similar properties.

- **Fuzzy Logic (FL):** This idea was founded as per ambiguity of the real world. For inference purposes, FL considers set membership values, which might be in the range of 0 to 1. From other words, the degree of fair in a statement can take on between 0 & 1, not only 0 or 1. (That is, true or untrue)

16.1.2 Unsupervised Learning (Cluster Analysis)

It classifies objects into concrete or amorphous groupings of related things. Objects in one cluster are typically comparable to those in other clusters of the same type, but unique from those in other clusters. Clusters are a kind of data compression since they may be seen as an integrated entity. Clustering can be accomplished in a number of ways.

- **A Partitioning Method -** The procedure begins with the creation of a seed set of partitions, with k being the desired final partition count. Then, to determine the most effective data partitioning, it employs a remotion technique called iterative remotion, which involves repeatedly moving objects across groups. To name a few, we have K-means, K-Medoids, CLARANS, & their improved variants.

- **A Hierarchical Method –** The approach begins by organizing the given data objects into a hierarchical structure. The method utilized to perform hierarchical decomposition determines whether it is top-down or bottom-up. To improve hierarchical clustering and make up for the inflexibility of merge or split, try checking item linkages at each partitioning or employing micro clustering before applying further clustering methods, like repeated relocation on the micro clusters.

- **Density-Based Method -** classifies objects according to their density. Clustering is guided by either a density function (like DBSCAN) or the density of nearby

items (or both, as in DENCLUE). Data clustering with OPTICS, a density-based approach, produces a more acceptable hierarchy.

- **Grid-Based Approach-** Clustering is performed on the grid structure using this method, after the object space has been quantized into a fixed number of cells (Statistical information grid cells (STING) is one example).
- **Model-Based Method-** With the utilize of techniques such as the EM algorithm, conceptual clustering (such as COBWEB), & neural network strategies, this method determines which model best fits the data within each cluster. Both grid-based & density-based, Wave Cluster & CLIQUE are examples of clustering techniques (e.g. SOM).

1.6.2 Prediction

Prediction models forecast continuously valued functions. Classification models can be used to determine if a person is a safe or dangerous candidate for a bank loan, and prediction models can be used to predict the amount of money a customer would spend on computer hardware based on their income & industry.

1.7 PROBLEM FORMULATION

The IoT is a decentralized kind of technology. Due to its decentralized structure, this network's key issues are security & routing. The investigation's focus is on improving IoT network safety. Because the network is decentralized, malicious nodes can easily penetrate it and launch active & passive attacks. The location shield is an active form of assault that slows down communications over the network. During a location protection attack, the malicious node will take the most circuitous path to deliver data, slowing down the entire network. The preceding research work's suggested solution is based on threshold delay planning. The rationale for the threshold delay results in an extremely low detection accuracy of malicious nodes. A new approach to rapidly identifying malicious nodes in a network is presented in this study.

1.8 RESEARCH GAPS

1. The previous methods have a significant delay and they are also not the lightweight solutions and thus require more resources.

2. VN attacks are still hampering the network speed despite the existing solutions and they effect the network performance.

3. Rapid & accurate detection and isolation of malicious nodes from the network is not guaranteed in the existing methods.

1.9 OBJECTIVES

1. To learn, analyze & implement various secure routing schemes used for detection of malicious devices on the basis of triggering version number attack in IoT & evaluate the effect of the attack on network operations.

2. To design energy level calculation technique for trust-based mechanisms.

3. To design a malicious node detection algorithm for multipath routing techniques.

4. To implement the proposed algorithm for the detection of the version number attack in an IoT network & comparing with existing methods of various parameters.

1.10 RESEARCH METHODOLOGY

In this research, we utilized a multi-step approach as include:

Pre-processing: - The basic goal of this method is to have a comprehensive list of all the possible types of attacks that malicious nodes can launch. A location protection attack that is now ongoing has a negative impact on how well the network functions. By using the node location method & trust-based strategy, the enemies are discovered.

Employ the RSSI technique: RSSI is a method for locating nodes. Using this technique, we can determine the anchor nodes' power state. This method of measuring is heavily utilized by the most popular wireless communication protocols. A medium's electromagnetic wave energy is modeled using this technique, which measures the strength of the received signal based on this property. Environmental variations, which are also detected as a function of distance, have a substantial impact

on RSSI approach. The RSSI technique distributes the frames throughout the network and other communication-related sensors. Additionally, the distance is calculated using the received RSSI values. After receiving RSSI from the undefined nodes, the beacon nodes continued broadcasting it. Receiving Signal Strength, abbreviated as RSS, is the sum of the sending signal's strength & signal's propagation loss, minus the signal gain status. RSSI range uses a propagation path loss experimental model to determine distance. The RSSI system uses route reply messages to communicate with nodes that have received a flood of beacons, network-wide control messages, or other communications. If the same beacon is responsible for both responses, it is assumed to be the localized mole. With the localization of sensor devices in the network, it is now feasible to describe the precise location of the sensor devices & detect rogue devices. Assume that (x, y) is where unidentified node D is located. There are fewer functional anchors in the network as a whole when 'n' is used. To calculate the location within range-based localization, utilize the following formula:

$$\begin{cases} \sqrt{(x-x_1)^2+(y-y_1)^2} = d_1 \\ \sqrt{(x-x_2)^2+(y-y_2)^2} = d_2 \\ \vdots \\ \sqrt{(x-x_i)^2+(y-y_i)^2} = d_i \end{cases}$$

$$A = -2 \times \begin{pmatrix} x_1-x_n & y_1-y_n \\ x_2-x_n & y_2-y_n \\ \vdots & \vdots \\ x_{n-1}-x_n & y_{n-1}-y_n \end{pmatrix}$$

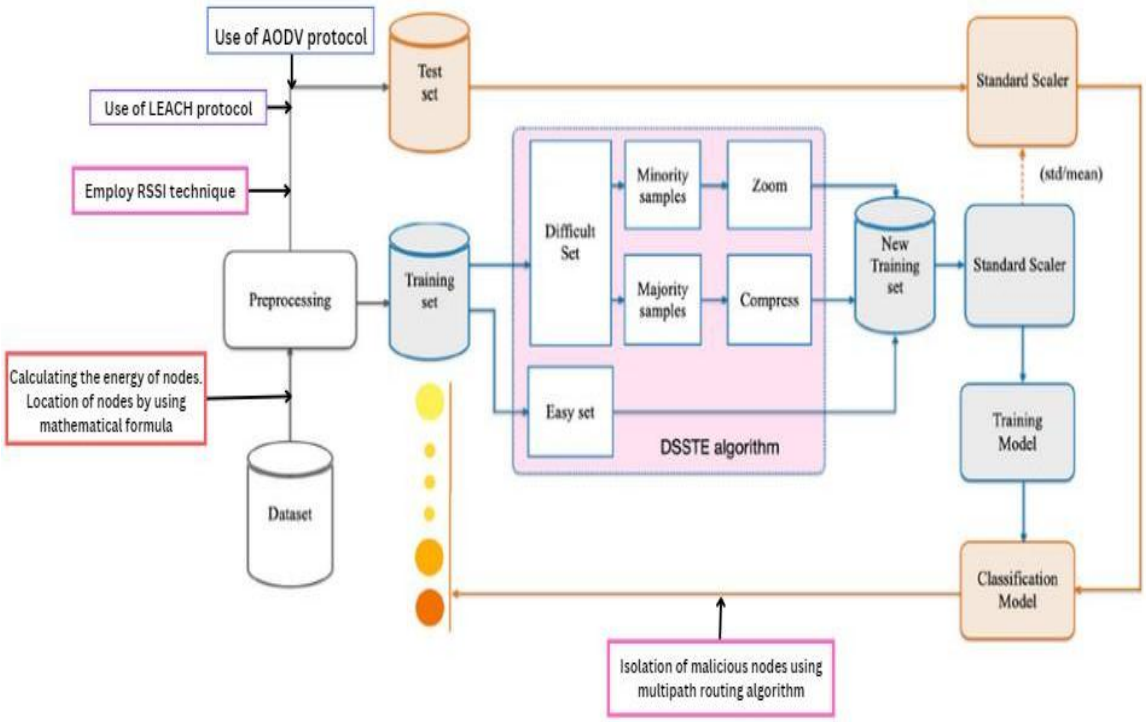
$$B = \begin{pmatrix} d_1^2 - d_n^2 - x_1^2 + x_n^2 - y_1^2 + y_n^2 \\ d_2^2 - d_n^2 - x_2^2 + x_n^2 - y_2^2 + y_n^2 \\ \vdots \\ d_{n-1}^2 - d_n^2 - x_{n-1}^2 + x_n^2 - y_{n-1}^2 + y_n^2 \end{pmatrix}$$

$$P = \begin{pmatrix} x \\ y \end{pmatrix}$$

Now, $P = (A^T A)^{-1} A^T B$

Trust-Based Mechanism: With the help of the trust-based mechanism, harmful gadgets can be identified. The energy of each node can be determined with this method. Choosing a sensor node with the lowest packet transmission and highest energy consumption allows one to identify it as a rogue node.

Malicious node isolation: Multipath routing isolates harmful devices from the network. The multipath routing algorithm does not choose a path that contains malicious motes. Estimates show that if you implement the technique, your network will last longer & perform better in terms of throughput, delay, & packet loss.



Overall representation

Chapter – 2

LITERATURE REVIEW

2.1 RELATED WORK

Ahmet Arş et al., (2020) concentrated on comprehending the influence of multiple VNA in RPL-based IoT networks [12]. The impact of many attackers was examined from a variety of perspectives. Based on simulations & analysis, it was shown that only the PDR was impacted by the maximum number of attackers, whereas neither the average network delay nor the average power consumption were. The outcomes demonstrated that the longer delays were occurred when the attacking positions were closer to the root and more efficient outcomes were obtained in terms of PDR during the central attacking positions by the higher power consumption. Finally, a recently proposed approach of mitigation was evaluated by calculating its effectiveness against a variety of attackers.

Salim Chehida, et al., (2020) intended an approach on the basis of attack-defence tree for assessing the applicable countermeasures to secure the infrastructure of IoT[21]. At last, the top of the statistical model checker was utilized for constructing anattack strategy exploration tool. The high impactful countermeasures were chosen by applying GA. The outcomes were valuable for highlighting the defence strategies ensuring a compromise among positive attacks, incurring the cost & time so that a sequence of attack actions was carried out. The experiments conducted over the IoT network attacks were also reported in this paper.

Ahmed Yar Khan, et al., (2020) emphasized on detecting the problematic scenarios in the IoT network with the help of AI [22]. In IoT contexts with limited resources, a lightweight strategy was proposed to identify insider assaults. This approach also had the potential to find anomalies coming from incoming data sensors. The results & comparison showed that, in terms of improving accuracy while detecting the attack, lowering the number of FP, & minimizing computing overhead, the proposed method

offered higher precision in comparison to the present approaches. The future directions plan to offer improved precision & insider assault attack protection.

Kashif Naseer Qureshi et al., (2020) A modern & protected framework built on RPL [27] should be used to identify security concerns in IoT & IIoT networks. To achieve this, the fundamental concepts of genetic programming were carried out. The recommended framework had potential for detecting a number of attacks such as VN, Sinkhole and Black hole attack. Several performance metrics were employed to compute the performance of recommended framework with respect to the precision, TPR and E2E delay. The encouraging outcomes were obtained from this framework and proved that the recommended framework was effective for IIoT environments based on RPL.

Max Ingham, et al., (2020) presented that the IoT had attained a lot of attention in the present era because of the flexibility, usability, diverse applicability and ease of exploitation [32]. This study emphasized on discovering the secure susceptibilities of IoT devices particularly which had employed the LPWANs. The vulnerabilities of LoRaWAN-based IoT security were examined, & vulnerabilities were identified. By utilising a model that predicts how the gadget will generate data, an attack was constructed and simulated. It was shown in the research that a jamming assault was initiated to prevent devices from successfully transferring data when the data creation model was predicted.

Ruchi Vishwakarma, et al., (2019) provided a study that was utilised to detect the infection using ML approaches [14]. To effectively & dynamically train the ML model, IoT honeypot produced data was implemented as a dataset. The suggested approach was considered as a productive outset to combat Zero-Day Distributed Daniel of Service attacks that had developed as an open challenge for the protection of IoT while dealing with the DDoS Attacks. As a future work, this approach would be prolonged further to the next level in which the real-time scenarios would be employed for discovering the open challenges or issues. A cloud server would also execute for tackling the Internet of Things devices constrained with resource.

Vijender Busi Reddy, et al., (2019) suggested a framework to reduce collusion & such attacks [15]. A novel similarity model was presented for quantifying the recommendation credibility at the place of direct trust which was employed as weight to evaluate the indirect trust. The affect of false recommendations was mitigated in computing the trust with the deployment of recommendation credibility as a weight to compute the indirect trust. The malicious motes were launched to assess the performance obtained from the recommended model. This model was proved efficient for the computation of true set of trust values. Thus, it was indicated that the recommended technique had potential for recognizing the malicious recommendations at the place of direct trust as recommendation credibility. For future work, the recommended model would supposed to be carried out on physical devices and conducting experiments.

Xupeng Luo, et al., (2019) stated that the SDN was implemented for designing the MTD model which assisted in the maximization of uncertainty due to the ever changing attack surface [16]. Furthermore, the IoT devices were impersonated, the attackers and malwares were tempted applying honeypots based on software-defined networking. Experiment results finally showed that SDN-based honeypots & Moving Target Defence could protect IoT networks from DDoS attacks & hide network assets from scanners. The effects of the presented method will be compared to those of other methods in the future as we deal with different types of attacks.

Meriem Bettayeb, et al., (2019) emphasized on developing a structure for an IoT security test bed which was capable for the analysis of the susceptibilities of IoT motes [17]. A road map & general technology were provided to set up the IoT Security Testbed. This framework included a test case template that assured test replication & standardized the test cases in addition to the hardware and software configuration. Moreover, a user-friendly platform was developed for the purpose of managing, running, and generating results from the test cases. A smart socket and smart thermostat were utilised to initiate three test scenarios in order to prove the feasibility of the proposed structure. The assessment results were then explained.

Arjun Shakdher, et al., (2019) described that the extensive penetration testing of IoT applications was carried out for finding out the susceptibilities [20]. Moreover, the OWASP was utilized to describe the limitations of the most susceptible security. The discovered susceptibilities were represented for testing a set of MIM attacks. The discovery showed that many IoT applications used in connected homes, security systems, healthcare, & automobiles were vulnerable to a number of attacks, and some of those applications had received over 1 million downloads. Recommendations for safeguarding the apps for IoT devices include countermeasures.

Ali Hameed, et al., (2019) analyzed that the IoT was easily seen at everywhere in these days. In smart cities, this technology was utilised for a wide range of purposes, such as controlling doors & air conditioners, preventing fires, and improving communication in hospitals, smart residences, & roadways [24]. IoT devices were related to the internet & used the network to send & receive a significant amount of crucial data. This led to open the starvation of attacker to occupy the IoT networks & stole its important data. The issue of IoT devices was that they had provided limited performance components due to which it became challenging to carry out the existing security technique on it. Thus, the lightweight algorithms were required to overcome these drawbacks that assisted the IoT devices. There were a number of suggested algorithms and authentication techniques reviewed in the survey for IoT for defending various types of attacks and taken into consideration the drawbacks of the IoT system.

Ning Zhang, et al., (2019) The downlink connection in IoT networks during probable pilot infection attacks, a RAV secure transmission mechanism was investigated [28]. The pilot sequences which were transmitted from an adversary and the receiver had not differentiated in this method and the sender employed that it received for estimating the CSI for beam forming or pre-coder design. After that, a random complex matrix was deployed to presuppose a set of data symbols so that the signal vectors were utilized to send. The proposed receiver was capable to recover the signal vectors via cooperation with a relay as proved by the means of security analysis. The typical outcomes showed that the adversary's BER was evaluated at 0.5 even though its channel quality suggested that perfect secrecy was offered.

Jalindar B. Karande, et al., (2018) discussed that the state-of-the-art was reviewed & computed for the techniques of detecting the security attack on Internet of Things protocol stack [18]. An extensive simulation was conducted to demonstrate the computation of the efficiency of this technique. The simulation outcomes showed that the attack detection technique failed in a state-of-the-art scenario when numerous attackers attacked an IoT system cooperatively. The simulated research also demonstrated how the network parameters in the IoT system significantly changed when several cooperative attackers began an attack outside of normal operations. The security attacks were detected earlier and efficiently using these deviations in parameters. This paper also described the necessity of the construction of an early detection algorithm to deal with the security attacks on IoT system.

Zie Eya Ekolle, et al., (2018) presented an approach utilized to deal with the security on Internet of Things for which the hybrid security strategy was implemented [23]. The described approach employs a clustering technique for unsupervised attack detection & grammar-based filtering mechanism to characterize a security protocols in opposition to IoT DDoS attacks. The presented security method was evaluated practically using emulation and the obtained outcomes were demonstrated. In future, the system would be optimized further for preparing it more effective to filter and detect the DDoS attacks. It would aim at establishing the system for detecting and preventing several diverse categories of DDoS attacks.

E. Abinaya, et al., (2018) addressed how the IoT made lives easier by creating a digital environment that was perceptive, flexible, & responsive to human needs [26]. One of the main sources of security vulnerabilities in a system's network was software attacks. This type of attack included viruses that purposefully put harmful code into the system and buffer overflows, which are DoS. Two encryption techniques, RSA & AES, were used to counteract the attack. The cryptool was used to carry out these algorithms, and it was also used to compute the performance of IoT devices.

Yair Meidan, et al., (2018) proposed N-BaIoT, an innovative method for the IoT that can collect snapshots of network behaviour & utilise deep auto-encoders to identify suspect traffic coming from infectious IoT devices [29].The popular IoT

botnets Mirai & BASHLITE, which compromised nine commercial IoT devices, served as inspiration for the suggested approach. Such points could be swiftly & correctly identified as the source of attacks according to the study's conclusions.

Ş. Okul, et al., (2017) described the significance of the concept of IoT in theoretical way. It was evaluated that although there was not any complete layer structure of Inter of Things, it had consisted of 3 common layers namely object, network and application layer [13]. Furthermore, Botnet, Social Engineering, DID, MIM and DOS were commonly known security epidemics at Internet on Things had discussed with illustrations and analyzed. At last, these attacks also defined the precautions which were essential in the layers of Internet of Things.

Jyoti Deogirikar, et al., (2017) analyzed that the IoT was theactive and discussed topic in present research field [30]. There were considerable research and development had carried out on IoT but, a variety of susceptibilities had seen that kept the IoT as a technology in danger. Consequently, various attacks were invented on Internet of Things prior to perform an actual commercial execution. The present study described the occurrence of several IoT attacks, their classification and counter measures and discovery of the most important attacks in IoT. A advanced study related to several attacks was presented and compared with respect to their efficacy and damage level in IoT.

Seungyong Yoon, et al., (2017) discussed that IoT devices were visible to various security susceptibilities & many security threats as these devices were resource-constrained and the interconnection of heterogeneous devices was done for offering a variety of application services [31]. Improving the safety & protection of IoT devices in an IoT scenario was the functional design objective of the remote security management server. The remote security management server offered & coordinated a variety of safety features in a streamlined and systematic way. As a result, numerous infringement situations that happened in the IoT environment were informed & prevented, quick and effective countermeasures were made possible to lessen the harm even in the event of a serious attack.

Ahmet Aris, et al., (2016) referred to how the RPL version number assaults were thoroughly examined & attack was analyzed from a number of angles [11]. An original feature of this work was the examination of realistic network architecture, which included both static & mobile nodes as well as various cardinalities. It was influenced by the IETF routing requirement specifications. Analysis had also been done on how version number attacks might affect how much power the motes used. Incorporating a probabilistic attacking paradigm, attacks were initiated by the attacker with a probability of p. The results of the performance were shown in terms of different values of p.

Lulu Liang, et al., (2016) discussed that an Internet of Things system was constructed as a target system. Kali Linux was employed with three diverse techniques to promotion the DoS attack [19]. The outcomes obtained from the experiment indicated that the association amid the sensor node and PC had great impact of DoS attack. The outcomes of the explored presented that the first strategy offered better performance & larger packet sizes. These three techniques were also compared. It was represented in the experimental outcomes that the highest performance was obtained from the first technique, the second technique performed more effectively in comparison with the third technique. For the future, the study of more attack techniques and the analysis of its affect would be carried out.

Mukrimah Nawir, et al., (2016) stated that a complex network having smart devices was involved in IOT that often exchanged the data via Internet [25]. The IoT's emergence as a cutting-edge technology paradigm that required the online storage of sensitive data & safety-critical operations made its security component very essential. The smart home, health care, & transportation industries all benefited greatly from network security. The shut down mode was activated when an interruption occurred while IoT devices were carrying out an operation. For the benefit of IoT developers, a taxonomy of security assaults in IoT networks was created to raise their knowledge of the dangers of security flaws. Better defenses were implemented in this approach.

2.2 REVIEW FOR INTRUSION DETECTION

In this chapter, various techniques for selection of intrusion detection as well as imperfection classification recommend during literature have been revised. Here the

researchers intended technique incorporated classification along with clustering methods. Several algorithms were expansively utilized for classification by several researchers during the literature review. The general effort is completed in the region of contribution onward soft computing techniques. Although several of the objects explored highlighted on the performance metrics, not several of highlights is going on categorization algorithms particularly for prediction. Additional exploration required to be approved with importance on preprocessing of data as well as classifiers purposely designed for avoiding imperfection prediction. It is supposed to require as determined to seek the explanation for the difficulty. In this chapter, the researchers also provide IDS research and shedding light on research directions. The researchers conclude the chapter with IDS research challenges identified and the proposals to answer these challenges.

Manish Kulariya, et al., (2016) used five ML techniques, including Naive Bayes, Gradient Boosted Decision Trees, Logistic Regression, & SVM. Also, in order to find and investigate intrusions in network traffic, a large data processing tool is used. Accuracy, prediction time, specificity, and sensitivity are used to evaluate experimental results on a real-time (KDD'99) dataset.

Bandre, et al., (2015) considered the architecture of a NIDS based on Hadoop . They improved performance by utilising seven IDS settings & General Purpose Graphical Processing Unit (GPGPU). Their research takes into account the Arpaio, DoS, & Tcp Syn attacks. In the end, they omitted to mention the system's accuracy.

Jabez & Dr. B. Muthukumar (2015) utilized a new outlier identification method in which the anomaly dataset was quantified using the Neighborhood Outlier Factor (NOF). The utilization of large datasets in a distributed storage system. The suggested IDS can detect nearly all types of assaults. The testing data's outlier values have an impact on the detection rate. The hypothetical dataset behaves like a set of incursion data if the outlier value rises. The findings demonstrate that related to other machine learning methods currently in use, the suggested system is faster at all levels.

Xiang et al. (2014) presented the ELM. Regression, multi-class classification, and binary classification issues have all been addressed using the ELM. For SLFNs, which have only one hidden layer, (ELM) is a supervised machine learning approach. They showed that a local (ELM) cannot process the entire (KDD99) dataset as effectively as their MapReduce variation of the (ELM) can.

Hu, W., Gao, J., and Maybank (2014) turned to two out-of-date online (GMMs) classifiers for the intrusion detection systems algorithms known as Adaboost & AdaBoost, which yield inaccurate findings. Also given is a distributed design in which each node constructs its own live parametric detection model utilising the Adaboost algorithm. Collaborating local parametric models that handle sparse data at each node allows for the construction of a global detection model. These objects are sorted into groups using SVM & PSO.

2.2.2 Soft Computing Methods

2.2.2.1 Machine learning, includes neural networks, SVM

Theyazn, et al., (2015) provide a cohesive approach that enhanced classification accuracy by identifying problematic packets and removing them from the system. This method's innovation lies in its utilize of non-crisp clustering techniques to describe ambiguity, specifically FCM & rough k-means. Also looked at were FCM & RKM clustering, which might be used to loosely or accurately determine the class of confusing packets. We display and discuss the contrasted outcomes of the SVM & J48 classifiers on two popular data sets.

Nilam Upasani, Hari Om, et al., (2015) utilized a fuzzy min-max NN. The user must provide a threshold (t) and a parameter (p), which determine the size of hyperboxes to be taken into account when choosing a test pattern as an outlier, during the testing phase. A common database is used for testing. The old approach improved recognition accuracy, but at the cost of longer callback times because another vote computation level with an O(k) serial time complexity was added during testing.

Takkellapati, et al., (2012) reported an IDS hybrid model combining K-means & poly kernel (SVM) techniques. (KDD99) was employed in the experiment as the

dataset. Information Gain & Triangle Area-based Modeling have been utilized for preprocessing and lowering the dimensionality of the data collection (KNN). Their hybrid model's average targets examine both legitimate and malicious packets. They revealed that their model had a low incidence of false positives & high percentage of accurate detection.

R. Shanmugavadivu et al., (2011) used the fuzzy inference method to create a fuzzy decision-making module. An improved method for detecting network intrusions, fuzzy rule learning, was employed to automatically determine a valuable set of fuzzy rules to be utilised with the inference methodology. The initial generation of the specific rules was mining the frequently occurring single-length items from both attack & non-attack data. The specified criteria were then made into fuzzy rules, which were then handed to the fuzzy system that categorizes the test data. The suggested method's performance was evaluated using the (KDD99) dataset, and the experimental findings specified that it is successful in identifying a variety of intrusions into computer networking.

R. Chen et al., (2009) projected SVM for IDS utilizing a basic set theory. They used RST to reduce the database's dimensionality (DARPA). Feature selection makes use of RST. The appropriate train & test phases then make use of SVM. By utilising the initial dataset, we compare the outcomes of the SVM methodology, SVM, RST, entropy, & SVM analyses. They came to the conclusion that the (SVM & RST) performed better.

Bharanidharan Shanmugam et al., (2009) they built a hybrid model that can recognize both anomaly and abuse attacks and is based on improved fuzzy & data mining approaches. Additionally, they used an improved Kuok fuzzy algorithm to generate fuzzy rules & create if-then rules that correspond to typical descriptions of security attacks. Furthermore, a fuzzy inference engine utilizing a Mamdani inference mechanism with 3 variable inputs has been deployed to speed up decision-making. This research made a number of advances, including an enhanced apriori algorithm for quicker rule generation and a lower query rate, as well as decreased functionality for quicker attack detection & fewer false positives. The suggested

model was evaluated for effectiveness using the 1999 DARPA dataset as well as the on-campus real-time networking environment, and the findings were reviewed.

2.2.2.2 Evolutionary Computation (EC), including Genetic Algorithms

P. Gupta, et al., (2019) established a method based on reliability factors for spotting & avoiding black hole attacks in MANET. The suggested method determines each communication node's dependability factor, and the packets are only passed to its nearby node when the reliability factor is high. After the simulation, the result indicated that by doing little modification in the protocol, better results are obtained in relations of throughput, PDR, & delay.

Gurung & chauhan, et al., (2019) presented a method to mitigate the effect on packet drop requirements of an intelligent gray hole assault. The present detection system is referred to as G-IDS abbreviated as a gray hole intrusion detection system consisting of gray hole intrusion nodes deployed in the MANET. MANET is a relatively unstructured wireless network that allows for the transmission of data using radio waves. The critical assumption of this network, which was significant that every node is sensitive to connections, however, in actual situations, individual respects could be hostile nodes as well as thus have to selectively drop of data packets rather than sending those packets up to the side of a destination. The model outcome represent the current schemes enhance the reliability of network based on PLR, PDR along with the average amount of throughput. The downside is now the proposed system is contrasted with the current scheme and demonstrates that the future scheme is more significant in relations of PDR, PLR, and Overall Performance compared with the current scheme.

Trivedi, et al., (2019) future a security mechanism in the identification of suspicious activities that occur within the network. This is possible using an improved data routing technique that is helpful in the identification of node that is suspicious. Many distinct types of attacks can compromise the majority of currently used MANET routing protocols. It might also be weakened by the well-known attacks of black holes and grayscales. When conducting a path discovery attack, malicious nodes pose

as vulnerable paths leading to a target node. When a lot of malicious nodes act together, this attack becomes complex.

Zardari, et al., (2019) DDBG is a novel method that has been developed to safeguard MANET from these types of attacks. The nodes are detected as black & gray hole attacks based on two conditions, such as energy and its absence in the blacklist. Also, the designed approach has been combined with a genetic algorithm (GA) to develop highly efficient IDS. The proposed approach DDBG performs better compared to the existing approach.

Brindha, et al., (2019) To combat the threats & increase security in the mobile nodes, a Fuzzy based secure multicast routing strategy was presented. The proposed Fuzzy decision system is utilized to identify whether nodes are normal or aberrant based on their authentication. To enhance the authentication of a node in the network topology, a different mechanism has been implemented in the multicast zone, including Signcryption or key generation principles. Using a network simulation tool, many parameters would show the FSMR performance, such as packet authenticity and average delay. Based on the simulation results, the presented mechanism accomplished enhanced performance based on certain parameters as compared to the previous schemes.

Jamaesha, et al., (2019) offered a protected location-aware routing protocol for the MANET routing mechanism's detection of the attacker node. Utilizing elliptic curve cryptography, a clustering mechanism was used to arrange the suggested work node within the network topology. The author showed how the suggested technique would use PSO to provide a future location. The optimisation of the node begins with the calculation of numerous characteristics, such as the link lifetime, speed, distance, & position. By factoring in the expected future locations of nearby nodes, we were able to determine a large node's trust value, which helped us detect rogue nodes in the network and reduce packet loss.

Debroy Saptarshi, et al., (2018) noted the difficulty of the second direction of the DSA. They have proposed the SPEED-IoT, an intelligent, power-efficient technique. Currently, a conduction-based power control overloading process is planned to

disperse the system's course demands without creating system-wide transmission overhead. They are examining the current state of the communication between IoT devices that employ these techniques. Additionally, they may analyze how their proposed plot would play out for various fundamental environments & IoT setups with respect to functional range, fundamental transmission characteristics, range properties, and the capacity for communicating with heterogeneous secondary IoT devices. They do not, however, reduce the effectiveness of security measures or transmission speed.

Xiaomin Li, et al., (2018) proposed a flexible deployment of the SDN Internet utilized by industrial (IIoT) computers. The IIoT serves as the building block for smart manufacturing & industry. IIoT is the process of effectively implementing various flow delay data that affects how well the system as a whole performs. Certain difficulties are addressed by the conventional plan. IIoT provides a fresh chance for development as a result of the significant development of SDNs, IWNs, & ECs. This article explores IIoT & associated domains in order to increase data transmission efficiency. The architecture for SDN & EC IIoT relays is suggested. The low-end time is currently supported by the coarse-grained transmission technique. By informing the PDD, an efficient transmission method is made available for all electoral procedures, & emergency scenarios, a well-designed system that uses a dynamic energy strategy is used. The suggested method is supported by imitation. The findings established that the proposed technique is a potential system for handling various data flows and, obviously, could lower the pillar IIoT, but it did not make use of the ML algorithm to get a better outcome.

Patel, et al., (2018) introduced the algorithm with a trust value as stated by or in IDS AODV which protects the network routing function against the attacker. The network does not have central control, and therefore, the use of a pairing method is considered to help detect node trust values. The cluster heading with the maximum confidence value was used by the researchers to choose the nodes. Each time a node directly upgrades a number of nodes by adding more packets. When a certain number of reliability nodes is equal to or equal to a node it can be considered as the true location

of subsequent connections. With this type of IDS, the network path can be saved from attackers.

T. A. Kolade, et al., (2018) have used Reply Supervision Technique (RST) in combination with AODV. This technique is utilized to find out the maximum sequence number in the RREP message along with the lowest hope count and hence detect the black hole node. A network has been built in which mobile devices act like nodes as well as regarded as a lesser infrastructure along with a wireless property of network having nodes communicate efficiently as well as it can also change their locations. Energy utilization was identified by evaluating the architectures with novel security management.The bandwidth utilization, along with a battery in case of any device, doesn't depend on the configuration of the threshold being correctly used. The attacks are identified named as Byzantine in case of recognized one is not handled in due time by any authority.

Jain, et al., (2018) demonstrated a recent mechanism to moderate the blackhole attacks in ad hoc AODV protocol because of the weighted binary relational fuzzy confidence model. For this analysis, the writers have utilized fuzzy-based Trust Computing (FTCP) approach as well as calculate the output for terms of PDR, percentage of node exhaustion of energy, amounts of node buffering as well as the number of calculations needed. During this function, the network output differs due to the variable number of hostile nodes. The result indicates that in terms of network efficiency measurements, the suggested method, FTCP worked stronger than the AODV and Sivagurunathan models.

Chhabra, et al., (2018) put out a game-theoretic security framework for protection against black hole attacks based on Potential Threat (PT). The proposed method frequently gained an advantage due to the reduced transmission of packets that delivered data punctually. It lessens the amount of energy that nodes use as well. The study revealed that the nodes were determined based on the list of ranks, which is the underlying issue. The lack of a game-theory mechanism that would enable nodes to transition between various states of energy is a limitation of this work. The outcome is a need for additional storage space due to the large number of rule sets.

Abdel-Azim, et al., (2018) created a fuzzy IDS that protects against black hole and gray hole threats simultaneously. The framework relies on an automatic method for defining the values of MFs parameters in the fuzzy inference section, in this case an ANFIS used to avoid errors in establishing parameter values in GA integration, throughout the intrusion detection system's function. In the event of low-speed stability, the rate of black hole assault is 37.8 percent, using the new IDS in the PDR for an improvement of 2.5 percent in the Routing Overhead, revealing an overall network growth of 36%. In the situation of low-speed stability, the rate of black hole attack is 37.8 percent, employing the new IDS in the PDR for an improvement of 2.5 percent in the Routing Overhead, revealing an overall network growth of 36%.

Persis, et al., (2018) The Firefly Method outperforms the exact approach in relations of both scalability & processing time. Routing in MANETs is a persistent graph problem that can be addressed using a number of different shortest path search algorithms. Modern routers use predetermined routing algorithms that filter out an undominated subset of possible solutions. The search efficiency of these algorithms has exponential temporal complexity, if the worst case scenario occurs. Furthermore, this MANET issue is a multi-objective optimisation challenge, suggesting that a fresh topology design would be necessary.

Kondaiah & Sathyanarayana, et al., (2018) have presented a secure route by using Firefly with PSO technique. The designed algorithm has created a route by adjustingthe acceleration coefficients in the rule set generated by the Firefly with the PSO algorithm. The network has been simulated in the NS2 simulator, & presentation has been measured in term of delay, detection rate as well as throughput in the presence of specified attacks.

Shona and Kumar, et al., (2018) have created an IDS system employing the idea of data mining to identify suspect nodes in MANET. The network performance has been enhanced by using the concept of Firefly with GA. As a classifier, RNNs rely on the features extracted from nodes using the hybrid optimisation approach. The results reveal that the attacker nodes are detected with less complexity in minimum time.

Balamurugan, et al., (2018) suggested a routing protocol of hierarchical nature that included the framework of data propagation with different amounts of transceivers. After the CH is designated by the network source connector, the clustering procedure is carried out in this work using the neighbor discovery algorithm. Therefore, the benefits of the mechanism include improved delivery ratio, drop, delay, higher throughput, & visible line of sight, which are produced by changing the number of transceivers. As we change the node amounts, we will compare our protocol to the current hierarchical cluster-based routing system.

Saqr Mohammed et al., (2017), there are two ways that need to be used to solve IDS issues. Feature extraction was done using Principal Component Analysis (PCA), & classification problem was solved using the Naive Bayes method. The model was evaluated using data from (KDD99).

Shiuan Meng Pan, et al., (2017) demonstrated a multicast-based regulation system that considered the physical location of IoT applications. When they succeed, they shorten the lengths of multi-part systems & reduce the number of source connections those systems require. Three stages—the solicitation, reform, and reform stages—comprise the plot. Support requests include information on how many additional hubs are needed to reach the final destination, and local hubs receive this information. Multicast modes can be reintroduced in the new refresh and transition phases via a reprogrammed update.

Huang Jun, et al., (2017) The multidimensional multi-constrained multicast steered problem can be significantly simplified using the proposed calculations, which use the entropy process to combine multiple constraints into a single exhaustive measurement. This allows for the use of well-known computations to find a solution. They have demonstrated a fictitious analysis of the complex structure & approximate of the suggested calculations, and they have carried out extensive simulations to gauge how the computation is presented. Explanatory & exploratory results have shown that one of the projected calculations is more efficient and precise than a delegate multi restricted multicast steering calculation.

Hasan, Mohammed Zaki, Fadi Al-Turjman, et al., (2017) put forth a particle multi swarm optimization (PMSO) method that is bio-inspired for creating, recovering from, & selecting k-disjoint multipath courses. The position information for both the optimal location for each individual & global location is supplied as a speed update to improve the display of directing computation. We analyzed the target function, which takes into account the typical energy consumption & typical in-organize delay, to accept this technique. Their findings demonstrate that the strategy utilizing all personal preeminent data is a valid method for improving the PMSO deployment. Additionally, similar estimates that enlarge the energy consumption & typical delay over the explored techniques have been compared to the suggested computation.

Khan et al., (2017) The unique CRB-MAC protocol has many desirable qualities, but it is power-hungry, relies on a limited set of nodes for next-hop delivery, and is vulnerable to disruptions in the network. Increasing the number of beneficiaries may not be the most cost-effective strategy for creating a sizable enough next-bounce collector. A device like this cannot have range detecting capabilities if the MAC protocol is based on the PRMA protocol. On the contrary, a door observes the separation. Although this method helps lessen energy consumption, it may still be incompatible with delay-sensitive programs. For MAC & guiding protocols in CR-based SG frameworks, they could also create & implement a common assessment structure.

Sharma Vishal, et al., (2017) Systems for "Energy Efficient Device Discovery for Reliable Communication on 5G-based IoT & BSNs Using Aerial Vehicles" IoT & BSNs were used to develop a technique for 5V-based UAVs in this research. The proposed method utilizes a power model to simulate optimal utilisation of the envisioned 5G-PPP of the future. Because of the high cost of the state network & restricted power available, a dynamic framework is created utilising XML charts for resource discovery. This invitation implies that power efficiency escalates to 78.4% while reducing delays by 21.3% without compromising delivery rates.

Meng Yue, et al., (2017). They talk about social interaction in D2D collaborative networks in this article. They initially provided the physical & social model of social

relations in order to foster social ties. Peer selection & cooperative network assessment are two of the primary issues that were looked into. Describe the works on the subject that are currently available in brief. The peer selection procedure in relationships with others was then developed using a network-based support network analysis technique. The outcomes of the imitations demonstrated the efficacy of the suggested techniques. They can use centralized authentication to protect their conversations.

Saudi, et al., (2017) proposed the effectiveness of some specified protocols of MANET named Dynamic Source Routing (DSR), AOMDV, DSDV, & AODV. MANET simulations have been undertaken to observe the behavior of all such protocols discussed, with the distinction between mobility as well as node frequency. MANET network is a compilation of wireless technologies that make up a network to communicate. Several routing protocol types were intended for MANETs.

Sethuraman et al., (2017) suggested a revised energy-efficient Trust-based routing technique called ReTE-AODV. This work must show that the suggested algorithm generates MANETs with an efficient path, despite their dynamic topology & openness. Instead of being transmitted through the shortest path, the packets are sent through the source to the destination after finding a trustworthy path that uses the least amount of energy and is trustworthy to send the packets. This work's simulation results show that it works better than the prior mechanism in relations of energy efficiency & trust-based routing. In this trust assessment example, the network's performance is evaluated. The main flaw is that it doesn't include further features for recognizing node behavior to the proposed model

Chandrashekhar Azad, et al., (2016) suggested a hybrid intrusion detection system that uses particle swarm optimisation in conjunction with fuzzy min-max neural networks. Using the cleaned-up (KDD99) dataset, we tested the proposed system extensively. Examining the rates of classification error & accuracy is one approach to evaluate the system's efficiency. We compare the proposed system to a wide range of classifiers, such as multilayer perceptrons, RBF, Fuzzy Multi-Modal & Fuzzy Multi-Modal Gaussian Automatic Classifier, and many more. According to the findings, the suggested system is better than the rest.

Qiu Tie, et al., (2016) carried out a study on ERGID: Effective routing protocol for evacuation reaction IoT. Response to an Emergency Shahad developed an IoT-based emergency response system called ERGID, which stands for Internet of Things Relying on Global Information Decisions. With the ERGID guiding protocol, they aimed to improve the Internet of Things system's proof of dependable information bundle transfer and effective critical situation response. By utilising an adjournment estimation-based component of the Delay Iterative Method (DIM), the problem with the systems not following acceptable and legal paths was uncovered. In addition, remaining Energy Probability Choice (REPC) is created with the goal of reorganising pieces in the IoT stack, taking into account the remaining energy of transmitting sensor hubs. When it comes to QoS criteria like end-to-end (E2E) delay, package loss, & power consumption, the anticipated ERGID steering protocol beats out EA-SPEED and SPEED, according to simulation results & evaluation of the proposed IoT architecture. Making judgements with partial information is inefficient when dealing with enormous amounts of data since people are often kept in the dark.

Bello Oladayo, et al., (2016) presented a method for intellectual device-to-device communication within the IoT. Direct-to-device (D2D) communication is an critical part of the IoT ecosystem for setting up, sending, and maintaining a suitable IoT biological system. Numerous concerns are currently being attended to by experts in the academic & industrial worlds. Energy productivity, management, security, context- awareness protocols, and other topics are included in some of the IoT research questions. The current focus has been on problems that affect direct D2D communication in an IoT setting. They dissected steering techniques & top-notch communication components in licensed & unlicensed wavelengths that can support shrewd D2D communications. They presented about how to fix these problems & highlighted some of the major obstacles that may derail introspective D2D communication in the IoT. Ultimately, they could recover their communication by using optimization computation.

P. Srivastava et al., (2016) By relating MANET with the Internet through Internet gateways, the use of the Internet is made better. Networked areas must have an Internet gateway for Internet connection. The traditional gateway comes close as easy

to use, efficient and the hybrid has a small throughput with startup sales. According to the novel gateway approach, the use of guided locales is attributed to the high reduction of the network through local adjustments acting as a traffic load on gateways. The composite approach has used the time element feature in addition to the line length and hip-hop calculation in selecting an enhanced online gateway. The researcher analyzed the proposed performance and timing of the gateway acquisition and navigation over the route. The results revealed that the research work showed improvement as compared to traditional methods.

Zhou, et al., (2016) presented random walk distance, which is calculated from other nodes using information retrieval concept and provided ranks to the nodes/ edges. The inputs are high scores' node alignments. The distance between each network's node pairs in the product graph of the two systems can be used to distinguish them from other high-scoring alignments. The developed model accurately implements the random movement concept, and the likelihood that energy consumption will be reduced will increase if the model is relay on a clustering device. A cluster-based energy efficiency secure model is used to solve the energy consumption issue.

A. Adnan, et al., (2016) suggested a novel Secure Routing Protocol (TESRP) in WSN, which utilizes a decentralized confidence framework in the identification and isolation of misbehavioral nodes. The Composite Routing Function (CRF) function, which is integrated into TESRP, calculates hop counts and residual energy using trusted nodes. The communicating route has taken the nodes with high CRF into account. The author utilized various loads and numerous estimated simulation metrics in this work. The simulation's outcome demonstrates that PDR is where black hole attack performance has fallen off the most, despite the fact that access points also block all data packets that would otherwise travel the path. Also, the efficiency of the throughput has significantly reduced with the creation of suspicious nodes.

Otermat et al., (2015) By taking use of empty lists and low-power Internet of Things devices, this limited bandwidth might sustain transfer speeds of up to 60.8 Mbps. Only through a combination of strong management systems & unwavering resolve can such pinpoint accuracy be attained. These in-depth analyses will find out whether

CR can increase the number of short-range IoT systems that use less power while decreasing the possibility of interference with licenced FM radio broadcasts (important clients). As a result of the board's regulations regarding the utilisation of force & obstruction, FM radio broadcasts are limited, making it possible for neighbouring IoT devices to share the available frequencies. Since 76% of the FM radio spectrum is presently unused, there is a great deal of room for innovation in how low-power, short-range IoT devices utilize CR. If there is indeed a sizable chunk of unlicensed FM radio spectrum in the US, it has to be thoroughly investigated. If we want all IoT devices to get along, have high throughput, and not interfere with FM radio broadcasts, we need to figure out how many devices can cover an imagined terrain with no FM radio interference.

Feng Zhiyong, et al., (2015) based on the proposed grid's distinct board base range (DSM). It has been demonstrated that it can minimize DTV frame interference while delivering dependable connectivity to major SG applications & optimizing for many systems. According to the requirements of the SG applications & QoS, the products are distributed. Additionally, there is a lot of use of empty Digital TV (DTV) teams to improve compatibility for these applications.

Wang Xiaofei, et al., (2015) introduced the Artificial Intelligence-based method to the Evolving Heterogeneous Network: Artificial Intelligence, Opportunities & Challenges. The items "X" work for are things that are addressed and demonstrated to be capable of developing & becoming competent in the best development of HetNets through this research & analysis. Additionally, they have highlighted both their technical challenges and exploration challenges along with their greatest advantages throughout the taxonomy. In particular, automation-based auto-optimization on a virtual interface coupled with SDN, M2M, and IoT is one of the most intriguing open research problems. The most advanced level of fantastic capabilities & most expensive price range of emerging HetNets will be displayed. HetNets can be a strong force for enhancing the performance of mobile services by including SON-designed SON features. The adaptor for detection's restoration, strong automation capabilities for system development, effective test execution, and fault

tolerance program execution account for the majority of contributions. By combining stages & optimizing, they can achieve the goal of an attack.

N. Arya, et al., (2015) proposes a novel algorithm that using the AODV routing protocol for examined and keep away from these two attacks named as wormhole collaborative black hole attack. The author demonstrated that avoidance helps improve the PDR as well as minimizes the power of overhead through enhancing routing protocol efficiency. The result shows that the throughput and PDR of Trusted AODV perform better in contrast to Worm Hole Attack and Collaborative Blackhole attack. Because the researchers haven't determined the trust value of the wormhole & collaborative black hole, which affects the network's results, this approach has a big drawback.

Zakaria, et al., (2015) executed queuing network analysis using firefly as an optimization approach & proved a secure way to migrate or transmit data to nodes in close proximity to one another is analogous to fireflies' attraction to brighter females. If no other node is more illuminated, the node will move at random. After the detection of the shortest route, the reply time has been determined by utilizing the queuing model. The response has been determined by drawing a graph between attractiveness and response time.

Patel, et al., (2015) have designed MANET in which the protection has been done against the packet drop nodes by utilizing the concept of SVM to categorize the approach. The classifier has been utilized to classify malicious nodes. Neighbor nodes trust value has been applied as an input to the SVM. The trust value has been computed based on the data packet & control packets. The SVM algorithm has been integrated with the AODV approach, and the results show higher PDR, throughput.

Mukesh, et al., (2015) have presented a feature selection-based IDS MANET system by integrating the nature-inspired based (GA) approach with the SVM algorithm. The nature of mobile nodes has been determined in the network layer. The irrelevant features have been reduced by using GA as an optimization approach and hence minimized the system complexity. SVM classification mechanism is trained with the

chosen features, and then it can be utilised for identifying malicious nodes by comparing them to data contained in its database.

Shubhangi Singh, et al., (2014) High detection rates have been achieved by combining GA & information theory techniques. A GA & (SVM) model were used to apply feature selection for classification. Various methods based on GA were used to acquire various detection rate percentages. As a result, findings for intrusion detection achieved utilizing GA approaches are superior to those produced using other soft computation methods.

Alah Eddine Benaicha, et al., (2014) introduce a GA with an improved selection operator & beginning population. The search for attack scenarios in audit files is made more efficient using GA. The reference dataset for (NSL-KDD) was utilized to find instances of misuse. By a high detection rate of 99% & low proportion of false positives (3%), the results are satisfactory. The outcomes are attained through a number of adjustments to the methodology, including the selection of the initial population for each sort of attack.

Ishino Masanori, et al., (2014) demonstrated a sparse & light multicast controlling standard for IoT applications. They have therefore shown that, when needed bundle conveyance rates are close to 0.9, our steering engineering can reduce Bloom Filter size to about 73 [Kbyte]. Additionally, they might be aiming to address the exploratory issues, such as how to reduce overheads and update teaching materials. For IoT applications, they suggested the multifunctional directing engineering employing Bloom Filters, and then they discussed the practicality of our steering system.

Nagaraju Devarakonda, et al., (2012) In this paper, the IDS model's creation utilizing a Bayesian network & HMM technique on the KDD99 dataset is discussed. Instead of 41 features, there are now only five. The framework was created to support many processing levels, including model learning using learning data & creation of a Bayesian network, which was then used to create a state transition diagram. The framework has undergone independent training & testing for connection records of a

normal & attack type. The outcomes demonstrate that the framework performs at a high level for the classification of regular and incursion assaults.

Hesham Altwaijry, et al., (2012) improved the R2L attack's accuracy by using Bayesian techniques. With a DR of 85.35% using the three characteristics: 23, 24, or 31, and a threshold value of 0.6, they achieved higher results for the R2L attack than Chou, who achieved 69.82% in his Ph.D. dissertation (Chou, 2007). The CR of 76.69% is also considered low when compared to Chou's result. Using a low threshold value reduces the accuracy of normal record detection (TN) and increases the DR of the R2L attack. An IDS system should utilize many Bayesian filters concurrently with each optimized filter to identify a particular type of record in order to increase accuracy.

2.3 ALGORITHMS USED

ALGORITHM 1

This study presents a reputation-based approach to identifying malicious nodes utilising the Improved LEACH routing protocol (MNDREL) [8]. Enhancing the efficiency of detecting malicious nodes in WSNs is the purpose of developing a new algorithm named MNDREL. Initially, the CH is chosen using the enhanced LEACH routing protocol. Once the nodes have been selected, the remaining nodes in the WSN can construct their own clusters and set the delivery routes for packets. Before sending a packet to the sink node, every node adds its own reputation-based evaluation value & unique identification in the last step. The sink node obtains the node numbers after parsing the packets. After that, it compiles a list of potentially malicious nodes by comparing these numbers to those of the nodes that originated from the source. In order to find the potentially harmful nodes in the network the fuzzy logic based multi-attribute trust model (FMATM) are all tested & evaluated. Results from simulated studies demonstrate that the MNDREL model outperforms competing models in detecting malicious nodes in WSN while simultaneously reducing the frequency of false alarms.

ALGORITHM 2

a) Data Extrapolation & Threshold

The suggested algorithm [22] begins by considering the experimentally harmful behaviour. Algorithm 1 presents the insider attack pseudocode. There are two sections to the procedure. The first involves loading libraries & dataset that contains sensor data. In a later stage, we'll choose one sensor and sort the data by various actions. Next, the data set is organised & shown on a weekly basis.

END Procedure

The second step is to smooth the data so that false positives are minimised. The foundation of the threshold algorithm are the three parameters: latency, threshold value, & influence value. The "threshold value" is the z-score at which the algorithm signals, "influence value" is the range of new signals on the mean & standard deviation, and "lag" is the lag of the moving window. To find out if a signal is positive or negative, data is evaluated by contrasting the standard deviation with the average value (mean). Apply the influenced value, modifying the filter, and lastly storing the resultant signal- based value constitute the last stage.

b. Distance Calculation Malicious

The distance estimate for any harmful activity is obtained in this step by bringing forward & processing data from the previous phase. The distance measurement is computed using the "stringdist" package, & loop is initialised with the dataset length. The sequential distance function is used when selecting an LV algorithm; following the first five weeks, which are deemed harmless, the resultant week value is incremented by +5. By analysing the week value with the distance value, we can tell if there is any malicious activity. Assuming no harmful activity has been identified, the method is repeated after making a little adjustment to the values of the Threshold Algo. The distance measurement method is repeated after rechecking the threshold algorithm's parameter "lag" and malicious activity week for different lag values. If necessary, the lag parameters can be adjusted incrementally. In Algorithm 2, we see the distance calculation pseudocode.

C. Benign/Malicious

When no harmful behaviour has been detected before, the last algorithm is used for benign activity. We run the Threshold Algo again after adjusting the parameters for benign activities. The next step is to run the algorithms that measure distance. Some have seen parallels between this and a modified approach that used different parameters in its original implementation. The pseudo code for determining if an activity is safe or harmful is shown in Algorithm 3.

ALGORITHM 3

Algorithm for static filtering

As a second-level packet filtering technique, the 6LoWPAN PFDL (Border Router) uses static filtering [23]. All external data traffic entering the IoMT network is filtered using PFDL Static. This filtering operation is applied to every data packet (Pi) in accordance with the source IP index that is predefined in the firewall ruleset. If the incoming packet is detected as a Backlisted IP during the SrcIP verification, it will be placed into the honeypot. We will verify if the packet size is within the limit if it is not. If the data packet's data size exceeds the data threshold, which is 120 bytes per second, it will be inserted into the honeypot. The packet will undergo frog GreyList IP verification before proceeding to threshold-based filtering.

ALGORITHM 4

Threshold based filtering

Within a specific time interval, all arriving packets undergo data size verification in threshold-based filtering. Figure 3 shows what happens when the same IP continues to send packets with the same data size over a certain time period; this pattern indicates that the IP in question may be the source of a denial-of-service assault. We assume that numerous sources are trying to perform a DDoS attack on a single medical device, therefore we divert their data to the honeypot & block the SrcIP from different sources. We can see that data is delivered at particular intervals from distinct Src IPs.

ALGORITHM 5

Encryption algorithms, including the AES algorithm & RSA algorithm, are used to fix the attack. The algorithms are tested on loT devices utilising cryptool, and their performance is assessed [25].

Rivest–Shamir–Adleman (RSA) is a prominent public-key cryptosystem for secure data transmission. Secure Sockets Layer (RSA) is based on the "factoring problem," or the problem of practically factoring the product of two large prime numbers. A solution to the RSA encryption problem has been found. We don't yet know if it's as challenging as the factoring problem [3]. Using a key big enough to circumvent the system has not been found in any reported techniques. One drawback of RSA is its slowness. This makes it an unpopular choice for direct data encryption purposes. The majority of symmetric key cryptography's shared keys are transmitted using RSA before they are utilizing for bulk encryption-decryption.

Advanced Encryption Standard (AES) algorithms Unlike Feistel cyphers, AES uses iterative encryption. The "substitution-permutation network" is its foundation. In some cases, it involves switching out inputs for certain outputs (substitutions), while in others, it includes rearranging bits (permutations). These processes are interconnected. With AES, a plaintext block's 128 bits are equivalent to 16 bytes. The arrangement of these 16 bytes is Bits aren't used by AES at all; instead, it uses bytes. A four-column, four-row arrangement is thus appropriate for matrix processing. The number of rounds is determined by the key length in AES, unlike DES. The number of rounds needed to implement AES with keys of 128 bits is 10, with keys of 192 bits it's 12, and with keys of 256 bits it's 14. Each iteration of AES uses the original key to generate a new 128-bit round key.

2.4 COMPARISON TABLE

Author	Year	Description	Result
Ahmet Aris, Sema F. Oktug, S.	2016	The unique aspect of this work is the analysis of realistic network topologies with variable cardinalities and mobile and static nodes as components.	The outcomes of performance were demonstrated in terms of several values of p.
Ahmet Arış, Sema F. Oktuğ	2020	A number of perceptions were considered for analyzing the impact of multiple attackers.	The outcomes demonstrated that the longer delays were occurred when the attacking positions were closer to the root and more efficient outcomes were obtained in terms of PDR.
Ş. Okul, M. Ali Aydın	2017	Botnet, Social Engineering, DID, MIM and DOS were commonly known security epidemics at Internet on Things had discussed with illustrations and analyzed	These attacks also defined the precautions which were essential in the layers of Internet of Things.
Ruchi Vishwakarma, Ankit Kumar Jain	2019	Suggested an approach based on honeypot in which ML methods were carried out for detecting the malware	The suggested approachwas considered as a productive outset to combat Zero-Day Distributed Daniel of Service attacks that had developed as an open challenge for the protection of IoT while dealing with the DDoS Attacks

Author	Year	Description	Result
VijenderBusi Reddy, Atul Negi,S	2019	A novel similarity model was presented for quantifying the recommendation credibility at the place of direct trust which was employed as weight to evaluate the indirect trust.	This model was proved efficient for the computation of true set of trust values.
Xupeng Luo, Qiao Yan, Mingde	2019	the SDN was implemented for designing the MTD model which assisted in the maximization of uncertainty due to the ever-changing attack surface	The Moving Target Defense along with honeypots based on SDN was capable to the hide the network quality from scanner & defending against DDoS attacks in IoT
MeriemBettayeb, Omnia Abu Waraga, Manar	2019	A road map was offered to setup IoT Security Testbed & overall technology to establish the testbed.	A smart socket & smart thermostat had employed to launch 3 test cases so that the feasibility of the suggested structure was represented
Jalindar B. Karande, Sarang A. Joshi	2018	An extensive simulation was conducted to demonstrate the computation of the efficiency of this technique.	The security attacks were detected earlier and efficiently using these deviations in parameters.
Lulu Liang, Kai Zheng, Qiankun Sheng	2016	Kali Linux was employed with three diverse techniques to launch the DoS attack	According to the results of the experiments, the first method yielded higher performance & bigger packet sizes.

Author	Year	Description	Result
Arjun Shakdher, Suyash Agrawal, Baijian Yang	2019	The OWASP was utilized to describe the limitations of the most susceptible security.	The discovery indicated that numerous Internet of Things apps in smart homes etc.
Salim Chehida, AbdelhakimBaouya, Marius Bozga, Saddek Bensalem	2020	Intended an approach on the basis of attack-defence tree for assessing the applicable countermeasures to secured the infrastructure of IoT	The outcomes were valuable for highlighting the defence strategies ensuring a compromise among successful attacks, incurring the cost and the time.
Ahmed Yar Khan, Rabia Latif, Seemab Latif	2020	It was proposed that a lightweight method could identify insider attacks & anomalies.	In terms of improving accuracy, the future technique outperformed the current techniques in relations of precision.
ZieEyaEkolle, KuramitsuKimio, Kohno Ryuji	2018	Presented an approach utilized to deal with the security on Internet of Things for which the hybrid security strategy was implemented	The presented security method was evaluated practically using emulation and the obtained outcomes were proved efficient
Ali Hameed, Alauddin Alomary	2019	The lightweight algorithms were required to overcome these drawbacks that assisted the IoT devices.	There were a number of suggested algorithms and authentication techniques reviewed in the survey for IoT for defending various types of attacks
Mukrimah Nawir, Amiza Amir, NaimahYaakob,	2016	To help developers of IoT devices better comprehend the dangers & constraints of security, a taxonomy of security assaults in IoT networks has been created.	Consequently, better protection was integrated

Author	Year	Description	Result
E Abinaya, K Aishwarva, Cecil Prabhaker	2018	The RSA and AES were two encryption algorithms which were carried out to rectify the attack	The computation of performances of Internet of Things devices was also performed.
Kashif Naseer Qureshi, Shahid Saeed Rana	2020	Recommended a new & protected framework for detecting the availability of security threats in IoT & IIoT networks based on RPL	The encouraging outcomes proved that the recommended framework was effective for IIoT environments based on RPL
Ning Zhang, Renyong Wu, Shenglan Yuan	2019	examined a RAV secure transmission technique to fend against potential pilot contamination assaults while preserving downlink connectivity in IoT networks.	The outcomes of the modelling demonstrated that the adversary's BER was only evaluated at 0.5, even if its channel quality indicated full secrecy.
Yair Meidan, Michael Bohadana, Yael Mathov	2018	Suggested a new network- based anomaly detection technique for the IoT known as N-BaIoT.	The assaults were launched via compromised IoT devices that remained a part of a botnet, which demonstrated the adaptability of the suggested method for detecting the attacks in an accurate & timely manner.

Author	Year	Description	Result
Jyoti Deogirikar, AmarsinhVidhate	2017	The present study described the occurrence of several IoT attacks, their classification And countermeasures and discovery of the most important attacks in IoT.	A state-of-the-art survey related to several attacks was presented and compared with respect to their efficacy and damage level in IoT.
Seungyong Yoon, Jeongnyeo Kim	2017	In a unified and well-organized fashion, the mobile security management server supplied and maintained multiple security features.	The IoT ecosystem saw a number of infringement situations that were preempted, and quick & effective countermeasures were made possible to lessen the harm even in the event of a serious attack.
Max Ingham, Jims Marchang, Deepayan Bhowmik	2020	This study emphasized on discovering the protecting susceptibilities of IoT devices particularly which had employed the LPWANs.	The study presented that the jamming attack was launched for blocking devices from sending data successfully when the data generation model was forecasted.

Chapter – 3

RESEARCH METHODOLOGY

3.1 PROBLEM FORMULATION

The IoT is a decentralized kind of technology. Due to its decentralized structure, this network's key issues are security & routing. The investigation's focus is on improving IoT network safety. Because the network is decentralized, malicious nodes can easily penetrate it and launch active & passive attacks. The location shield is an active form of assault that slows down communications over the network. During a location protection attack, the malicious node will take the most circuitous path to deliver data, slowing down the entire network. The preceding research work's suggested solution is based on threshold delay planning. The rationale for the threshold delay results in an extremely low detection accuracy of malicious nodes. A new approach to rapidly identifying malicious nodes in a network is presented in this study.

3.2 ALGORITHMS AND DESCRIPTION

ALGORITHM 1:

In this we discuss about two mitigation techniques with different resource requirement and performances.

- **Elimination (Mitigation Technique I):**

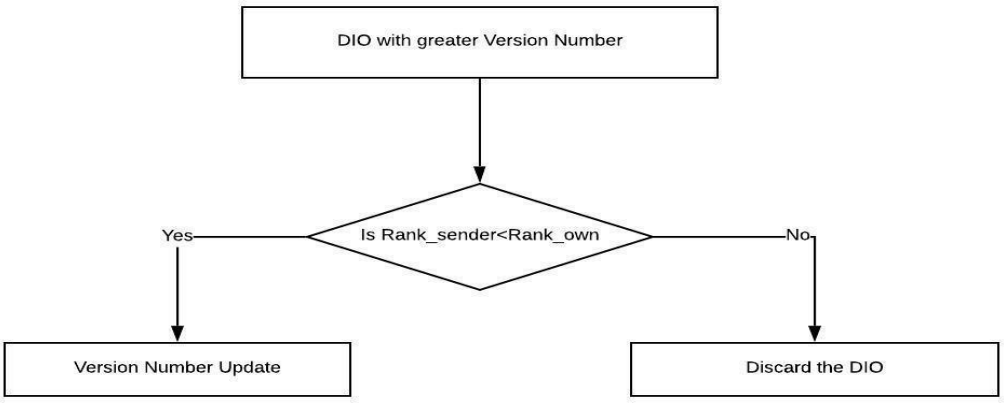


Figure 3.1: Elimination technique.

In the initial technique, a node take up Version Number updates commencing from the root of the DODAG to the leaf node.

The node will determine whether an update is coming from a node that is closer to the root nodes than it is after receiving a message with a higher version number, and if so, it will update the version number otherwise, as depicted in Figure 3.1. Here, the rank information is used to signify the position of a root node relative the RPL network, thus the VN updates are eliminated if the node which is sending the update had a better rank than a node which is receiving the update.

This technique can be executed in simple way by applying just one additional comparison operation, thus the devices which have limited resources can be easily deployed by the help of this technique. The network performances is effected more in terms when the VNA is smeared from such positions which are closer to the leaf nodes.

- **Shield (Mitigation Technique II)**

In earlier research, a shield-based solution was put out to isolate version number attacks on the DODAG protocol. The reactive criteria for path selection from source to destination form the foundation of the DODAG protocol. The source node initiates a deluge of route-request packets in the DODAG method, and nodes that are able to reach the destination send back the DIO message. The DIO message specifies the nodes with the greatest version numbers, & they are used to send data from the source to the destination. The bad nodes update their version numbers so they can send as much data as possible. The shield technique, is designed to isolate the malicious nodes which are changing the version number. The main assumption of the shield technique is that do not trust any of the node for the version number calculation. To follow the main assumption, in the shield technique every node calculate rank of itself and its adjacent nodes. Every node will maintain the shield list of its adjacent node which has better rank. The node stores the identification information like IP address and VN information. Assume that the network has four nodes, namely nodes A, B, C, & D. When compared to nodes A, B, & C, node D gets the highest rank. Thus, the VN of A, B, & C are stored in Node D. The DIO message

which is coming for the path establishment is analyzed using the shield list. The shield list which is maintains on every node can be updated according to modification in the network topology.

The mobility model that the network's sensor nodes use determines the topology of the network. The shield based technique final version number is calculated by combining various Shield list which are maintain by the various sensor nodes. In the shield based mechanism two conditions are exits. The two conditions are explanted below:-

1. **When the VN information is same that is maintain in the Shield List:-** The nodes maintain shield list of the adjacent node which contain VN information. When the nodes reply back with the DIO message which contains the VN information, if the VN information in DIO message get matched with the information in the DIO list then the path will be established from the source to destination.

2. **When the VN information is different that is maintained in the Shield List:-** In the second condition, when the VN information in the DIO message get mismatched with the information in the shield list. In mismatched condition two conditions get raised either to discard the DIO message or update shield list information according to DIO information. To update the shield list information multiple shield list are collected from the different nodes. The majority is calculated and based on the majority shield information get updated. The shield information updating procedure is explained in the flowchart given below:-

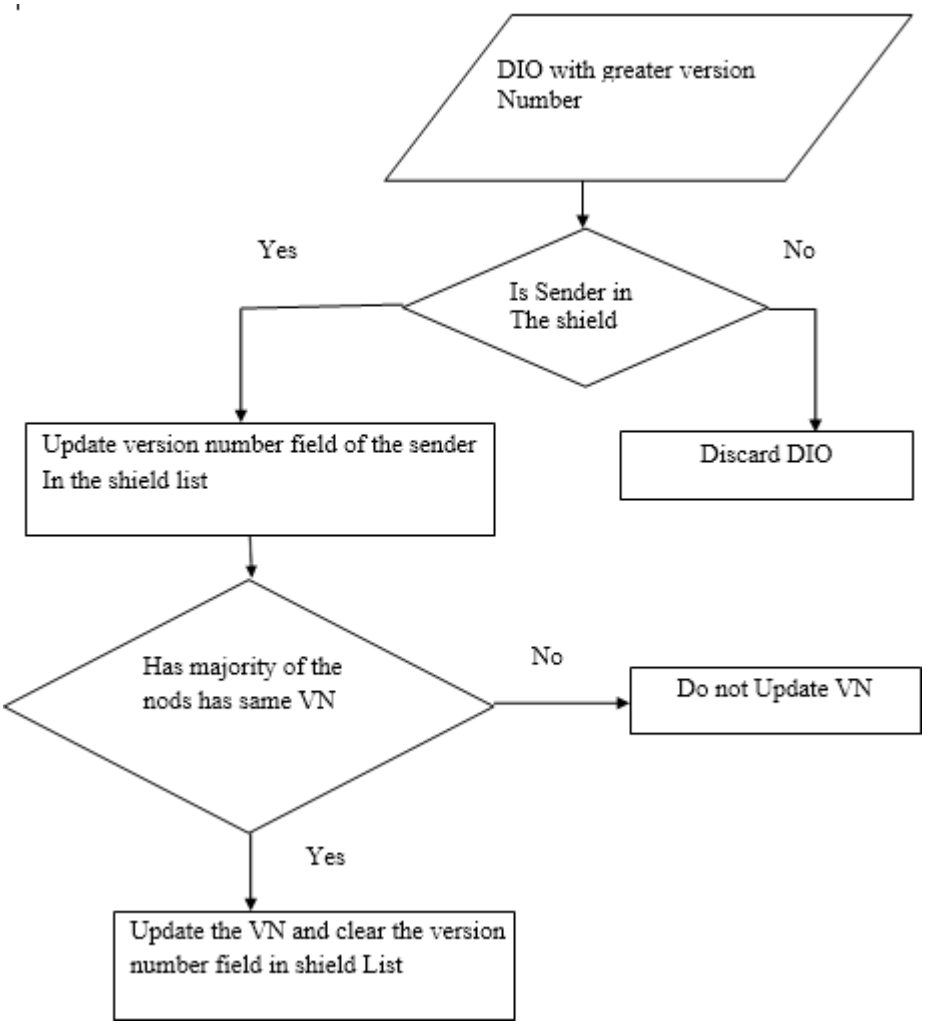


Figure 3.2 Shield Technique

ALGORITHM 2

(1) **Pre-Processing:-** In the IoT, there will always be the same number of sensor nodes. The locality-based clustering method is used throughout the whole network. Using the proposed LEACH protocol, we can reliably verify the strength and distance of each node. The most potent and centrally located node in the cluster is particular as the CH. All of the network's nodes will feed data to the cluster leader. The node coordinates with other nodes to build a path and reports its progress back to the home base. By utilizing the AODV routing protocol, a connection can be established between the origin and the final destination. The route reply packets are flooded by the AODV protocol at the

source node. Based on the highest sequence number & hop count, the starting node chooses the optimal path to the end destination. The first node uses a relay to send data to the second node. By following the chosen path of the intruding node, a version number assault can be launched.

(2) **Detection of malicious nodes:** - Multiple strategies have been established over the previous few years for identifying malicious nodes. During the prior method, we employed the monitor mode. The nearby node's activity can be observed with the use of this method. The method fails miserably at pinpointing malicious nodes. Earlier investigations have made use of a second strategy, the delay tolerance method. To locate attacker nodes, supplementary gear & software are needed for this method. Because of this, the plan has become more intricate and costly. The node localization strategy is utilized by the base station to determine & isolate the offending nodes. The data for node localisation is gathered along a set path. By utilising the node localization technique, every sensor node is able to transmit data to the base station. The base station is able to learn the precise positions of the sensor nodes as well as the time it requires for them to relay data to the rest of the network. When there are limitations on the quality of service, the base station investigates them carefully. The base station starts looking for the intruder node as soon as the network throughput falls below a certain threshold. The central node of the network detects intrusions by monitoring data transfer rates at each stage. When the throughput falls below a specific threshold, the offending node is recognised. Information obtained includes the distance from each node to the home base. The distance adds delay to each hop count along the final path. The base station has detected the delay. The delay at each node is calculated to find the culprits in slowing down the network and pinpoint the attackers. The predicted latency can be estimated with the help of Equation 1.

$$\text{Expected Delay} = time\ to\ live^{*} \frac{Distance\ between\ each\ node}{Distance\ between\ source\ and\ destination} \quad \text{--- (1)}$$

The distance between each node is considered with the equation number 2.

Distance =(a(i+1)-a(i))^2+(a(y+1)-a(y))^2 -- (2)

Equation number 3 defines the anticipated delay.

$$Predicted\ delay = \frac{\text{Distance between each node}}{\text{Total number of message exchange}} \quad \text{--- (3)}$$

An attacker node can be pinpointed by the network when the calculated delay is longer than expected. The critical delay is the time that you expect to wait. The maximum acceptable network latency is 2 milliseconds. If the sensor node causes a delay greater than 2 milliseconds, it will be identified as the attacking node. The multipath routing mechanism is used to disconnect malicious nodes along the path. The proposed approach is founded on relevant approaches & threshold mechanism for locating adversary nodes. This method is preferred for locating malicious nodes, and it can be implemented with no additional hardware or software.

(3) **Isolation of Malicious nodes: -** Eliminating all contact with the bad node is the last stage in creating a barrier between the source & destination nodes. By implementing multipath routing, the troublesome node is eradicated. The network successfully isolates the malicious node by utilising multipath routing. Any node in the vicinity of the destination will respond to route request packets transmitted by the source by sending a route reply packet. By considering the number of hops and the current sequence number, the starting node finds the best path. The targeted node is the one with freedom of movement, while the source node is not.

ALGORITHM 3

The identification & eradication of IOT intrusion are the main goals of this effort. In this intrusion, an adversary node establishes a route from source to destination after making its entry in the network. It's called a "black hole attack." In this incursion, the source node floods the network with RREQ packets. The nodes that have a route to the target respond by issuing RREPs. The enemy node falsely claims to have a direct

path to the target, whereas in fact it has no such pathway. This node also lies about having maximum sequence number. Therefore, the source node selects a path across malevolent node. The following procedures are used to identify the network's malicious nodes.

1. The IOT's moveable nodes are installed in the first step.
2. In order to ensure information sharing inside the network, the second phase identifies source & destination nodes.
3. The source node floods the network with fake RREQ packets.
4. The fake RREQ message-reverting node will be recognized as an adversarial node.
5. In the alternative scenario, the source node will track the RREPs' reception times.
6. The least trusted node is the one that responds with the smallest sequence number in the shortest amount of time.
7. We'll call this node the adversary node.
8. In this step, the clustering method is put into practice to remove hostile nodes from the network.
9. Based on location-based clustering, the entire network will be divided into groups.
10. From each cluster separately, the maximally trustworthy node will be chosen to serve as the CH.
11. The elected CH will do the routing of data to the destination by isolating the adversary node from the network.

3.3 OBJECTIVES

1. Recent reviews have shown that presently ML using IDSs do not fulfill the specifications for different scenarios.

2. This boosts the present work which aims to make network IDS designers imply Deep Learning techniques such as RNN, Deep-NN, & designing algorithms.

3. The will is to create a format that defines IDS design's ability to detect attacks

4. The focus is to make an effective computer network protection intrusion detection system focused on deep learning techniques.

5. It would be a continuous learning model based on a node's specific network parameters and will anticipate new attacks based on the shift in the behavioral pattern of network traffic.

3.4 RESEARCH METHODOLOGY

In this research, we utilized a multi-step approach as include:

Pre-processing: - In this case, the malicious node claims a direct route to the target yet has none. A location protection attack that is now ongoing has a negative impact on how well the network functions. By using the node location method & trust-based strategy, the enemies are discovered.

Employ the RSSI technique: RSSI is a node-location scheme. This method evaluates the power status attained by the anchor nodes. This method of measuring is heavily utilized by the most popular wireless communication protocols. A medium's electromagnetic wave energy is modeled using this technique, which measures the strength of the received signal based on this property. Environmental variations, which are also detected as a function of distance, have a substantial impact on RSSI approach. The RSSI technique distributes the frames throughout the network and other communication-related sensors. Additionally, the distance is calculated using the received RSSI values. After receiving RSSI from the undefined nodes, the beacon nodes continued broadcasting it. To calculate distance, RSSI ranging employs an empirical model of propagation path loss. The RSSI system uses route reply messages to communicate with nodes that have received a flood of beacons, network-wide control messages, or other communications. If the same beacon is responsible for both responses, it is assumed to be the localized mole. With the localization of

sensor devices in the network, it is now feasible to describe the precise location of the sensor devices & detect rogue devices. Assume that (x, y) is where unidentified node D is located, and (xi, yi) is where receiver for the i'th anchor node is meant to be. For unidentified devices, the di is taken into account as the distance among the target node & i'th anchor nodes. Additionally, 'n' degrades the overall number of functional anchors inside the network. The formula given as follows is used to compute the location inside range- based localization:

$$\begin{cases} \sqrt{(x-x_1)^2+(y-y_1)^2} = d_1 \\ \sqrt{(x-x_2)^2+(y-y_2)^2} = d_2 \\ \vdots \\ \sqrt{(x-x_i)^2+(y-y_i)^2} = d_i \end{cases}$$

$$A = -2 \times \begin{pmatrix} x_1-x_n & y_1-y_n \\ x_2-x_n & y_2-y_n \\ \vdots & \vdots \\ x_{n-1}-x_n & y_{n-1}-y_n \end{pmatrix}$$

$$B = \begin{pmatrix} d_1^2 - d_n^2 - x_1^2 + x_n^2 - y_1^2 + y_n^2 \\ d_2^2 - d_n^2 - x_2^2 + x_n^2 - y_2^2 + y_n^2 \\ \vdots \\ d_{n-1}^2 - d_n^2 - x_{n-1}^2 + x_n^2 - y_{n-1}^2 + y_n^2 \end{pmatrix}$$

$$P = \begin{pmatrix} x \\ y \end{pmatrix}$$

Now, $P = (A^T A)^{-1} A^T B$

P is signifying the coordinates of the sensor nodes.

Trust-Based Mechanism: The trust-based strategy is used to identify the malicious devices. Finding out how much energy each node has is a simple using this method. The sensor node that uses the most energy while transmitting smallest packets has been identified as the malicious node.

Malicious node isolation: Multipath routing isolates harmful devices from the network. The multipath routing algorithm does not choose a path that contains malicious motes. Estimates show that if you implement the technique, your network will last longer & perform better of throughput, delay, & packet loss.

METHODS

To improvise the IDS classification accuracy in the face of unbalanced network traffic, we offer the Difficult Set Sampling Technique (DSSTE) algorithm, which compresses the majority samples while increasing the number of minority samples in difficult samples. This approach reduces imbalance in the training set. When building our classification models, we include RF, SVM, XGBoost, LSTM, Mini-VGGNet, & AlexNet among our classifiers. Figure 3 shows the intrusion detection model that we suggested. Processing for duplicates, outliers, & missing values were among the initial steps in our IDS data pre-processing. After that, we used our suggested DSSTE algorithm to balance the data in the training set after splitting it from the test set. In order to expedite the convergence process, we utilise Standard Scaler to standardise the data & digitise the sample labels prior to modelling. The last step is to train the classification model utilizing the processed training set. The model is then tested utilizing the test set.

A. DSSTE ALGORITHM

Due to their similar representations, classifiers have a hard time learning to distinguish between minority attacks—which can lurk within a large amount of normal traffic— and other types of traffic data in the event of a network imbalance. Duplicated noise data makes up most of the comparable samples in the imbalanced training set. The classifier is unable to learn the distribution of the minority class due to the large quantity in the majority class, therefore we reduce its size. The minority group is defined by distinct continuous characteristics rather than static discrete ones. Consequently, the continuous characteristics of the minority group are amplified to obtain data that corresponds to the true distribution.

This is why we advocate for a fair playing field by employing the DSSTE approach. Partitioning the training set into a near-neighbor set & far-neighbor set using the

Edited Nearest Neighbor (ENN) algorithm is the initial step in correcting an imbalanced set. Due to the great degree of similarity between the far-neighbor set and the near-neighbor set, we refer to the former as "easy samples" and the latter as "difficult samples" when discussing the difficulty of training a classifier to distinguish between the two sets. Next, we highlight and blur the problematic set's minority samples. In order to create this new training set, we merge the easy set with the tough set's minority and their augmentation samples. All things considered, the K neighbors constitute the scaling factor of the ENN method. As the scaling factor K increases, we see an increase in the number of problematic samples, compression rate for most samples, & synthesis rate for minority class. One name for the DSSTE algorithm is algorithm.

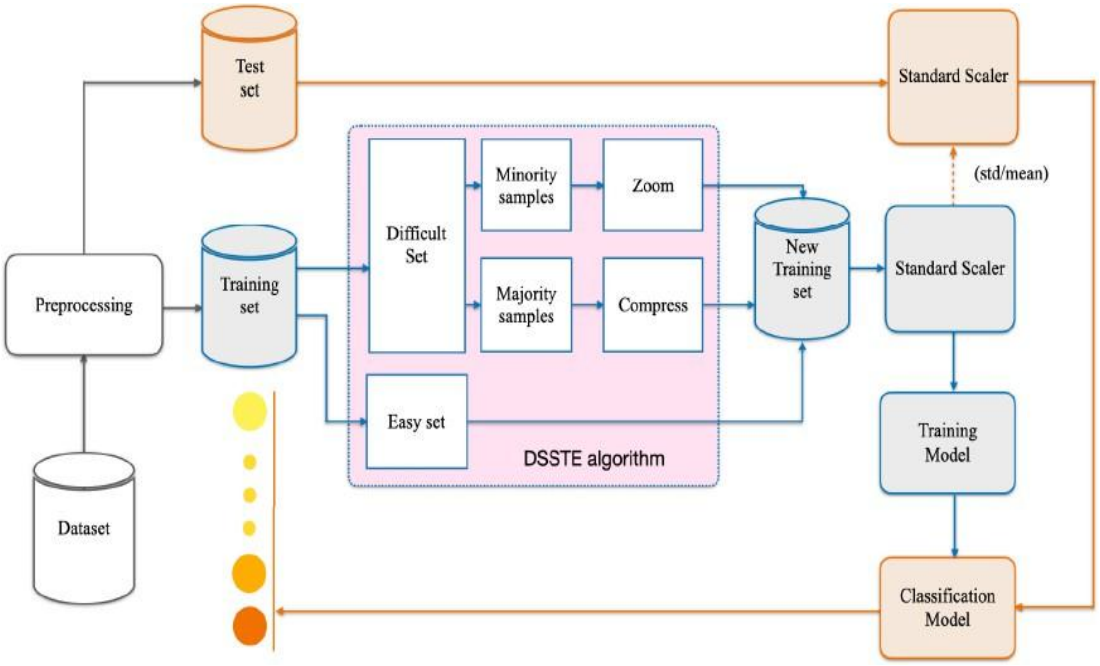


Figure 3.3 The overall framework of network IDS.

Algorithm 1 DSSTE Algorithm

Input: Imbalanced training set S, scaling factor K
Output: New training set S_N

Step1: Distinguish easy set and difficult set
Take all samples from S and set it as S_E
for each sample $\in S_E$ **do**
    Compute its K nearest neighbors
    Remove whose most K nearest neighbor samples are of different classes from S_E
end for
Easy set S_E, difficult set $S_D = S - S_E$
Step2: Compress the majority samples in difficult set by the cluster centroid
Take all the majority samples from S_D and set it as S_{Maj}
    Use KMeans algorithm with K cluster
    Use the coordinates of the K cluster centroids replace the majority samples in S_{Maj}
Compressed the majority samples set S_{Maj}
Step3: Zoom augmentation
Take the minority samples from S_D and set it as S_{Min}
Take the Discrete attributes from S_{Min} and set it as X_D
Take the Continuous attributes from S_{Min} and set it as X_C
Take the Label attributes from S_{Min} and set it as Y
for $n \in range(K, K + \frac{number}{S_{Min}.shape[0]})$ **do** // zoom range is $[1 - \frac{1}{K}, 1 + \frac{1}{K}]$, $S_{Min}.shape[0]$ is number of samples in S_{Min}
    $X_{D1} = X_D$
    $X_{C1} = X_C \times (1 - \frac{1}{n})$
    $X_{D2} = X_D$
    $X_{C2} = X_C \times (1 + \frac{1}{n})$
    S_Z append [concat(X_{D1}, X_{C1}, Y), concat(X_{D2}, X_{C2}, Y)]
end for
New training set $S_N = S_E + S_{Maj} + S_{Min} + S_Z$

B. ML & DL ALGORITHMS

The following section details the classifier's design & training and testing methods used: RF, SVM, XGBoost, LSTM, AlexNet, & Mini-VGGNet.

1) RANDOM FOREST

In 2001, Leo Breiman put forth the random Forest proposal. One of the best supervised learning algorithms, RF can be utilised for training a model to predict, given a dataset's characteristic traits & classification outcomes, which sample type a

certain sample belongs to. A decision tree forms the basis of RF, which uses the Bootstrap aggregating approach, also known as Bagging, to generate several training sets.. Weak classifiers are the decision trees that are generated; a robust classifier is generated by many weak classifiers; & input samples are classified using the voting process. When a fresh set of samples is fed into a RF, which has already trained a large number of DT according to a predetermined random rule, each tree in the forest independently predicts what the new set of samples will be. Then, to achieve the final result, the trees' predictions are integrated.

2) SUPPORT VECTOR MACHINE

It was initially proposed in 1995 by Coretes & Vapink. In high-dimensional, nonlinear, & small-sample pattern recognition, it demonstrates numerous unique advantages; it may also be applied to other tasks, such function fitting in machine learning challenges. Prior to deep learning's ascent, SVMs were the gold standard in machine learning. Vapnik Chervonenkis's (VC) dimensional theory of statistical learning & structural risk reduction principle form the basis of the SVM method. The main aim is to better distinguish distinct categories by finding a separation hyperplane between them. If the support vector can be located, the SVM approach will assume that the hyperplane can be established by separating the sample points that are closest to it.

3) XGBoost

A parallel RT model developed by Chen and Guestrin, XGBoost integrates the Boosting concept with an enhanced gradient descent decision tree. The XGBoost model circumvents the GBDT method's sluggish calculation performance and erroneous output. Regularization to the conventional GBDT loss function is included in XGBoost to avoid the model from being overfit. The classic GBDT uses the computed loss function to run a first-order Taylor expansion, with the negative gradient value serving as the model's residual. But XGBoost makes sure the model is accurate by running a second-order Taylor expansion. In addition, XGBoost sorts and blocks each feature, which allows for parallelization of the calculation when searching for the optimal split point, greatly increasing the calculation performance.

4) Long Short-Term Memory

As a framework for RNNs, the LSTM network was introduced by Hochreiter and Jurgen in 1997. An LSTM network, like the majority of RNNs, is universal since it can calculate all network elements that can be computed by any regular computer given the correct weight matrix. Unlike the more traditional RNN, LSTM network is great at learning from past mistakes. We can classify, assess, & predict the time series when we don't know the exact length or boundary of the lag between major events. LSTM outperforms competing sequence learning algorithms, such as hidden Markov models & RNNs, and is insensitive to gap length. The introduction of the gate structure & storage unit solves the problem of gradient disappearance and gradient explosion.

5) AlexNet

AlexNet is a well-known example of a simple deep learning network. In 2012, it was suggested by Hinton & Alex Krizhevsky, one of his students. An 8-layer deep neural network, excluding the activation & pooling layers, forms its primary structure. This network consists of 5-layer convolutional layers & 3-layer fully connected layers. In the convolutional layer of AlexNet, the activation function is the ReLU function, as opposed to the frequently utilised Sigmoid function in earlier networks. The issue of gradient dispersion in deep neural networks is addressed by introducing the ReLU function. To down-sample the feature map produced from the convolutional layer, the AlexNet neural network employs the Maxpooling approach rather than the more common average pooling. This means that compared to its predecessor, the AlexNet neural network performs better.

6) MINI-VGGNet

An innovative deep CNN called VGGNet was created in 2014 by academics from Oxford University's Visual Geometry Group & Google Deep Mind. The network took second place in the ILSVRC2014 classification project. In their work titled "Very Deep Learning CNN for Large-Scale Image Recognition," the importance of CNN depth in determining the accuracy of recognition for massive image sets is primarily discussed.

A tiny convolution kernel (3×33×3) is utilised to build different depths of CNN architectures, which is the primary contributor. The evaluation of these network architectures led to the conclusion that a network depth of 16–19 layers was optimal for improving recognition accuracy. When extracting characteristics from images, VGG- 16 & VGG-19 are often favoured. To some extent, VGG is like a more advanced version of AlexNet. A fully connected layer & convolutional layer are placed on top of the entire network. A small-sized convolution kernel (3×3) is used by VGGNet, in contrast to AlexNet.

Among the original, foundational networks in deep learning is AlexNet. In 2012, it was suggested by Hinton & Alex Krizhevsky, one of his students. An 8-layer deep neural network, excluding the activation & pooling layers, forms its primary structure. This network consists of 5-layer convolutional layers & 3-layer fully connected layers. In the convolutional layer of AlexNet, the activation function is the ReLU function, as opposed to the frequently utilised Sigmoid function in earlier networks. The issue of gradient dispersion in deep neural networks is addressed by introducing the ReLU function. In place of the more conventional average pooling, the AlexNet neural network's convolutional layer employs the Maxpooling approach to down sample the feature map output. This means that compared to its predecessor, the AlexNet neural network performs better.

With fewer traffic characteristics to work with, we ran our classification experiments using the Mini-VGGNet (miniVGG) network that Ismail suggested [34]. Two sets of CONV D=> RELUD=> CONV D=> RELU D=> POOL are typical components of Mini-VGGNet. Then, there is the FC= >RELU D=> FC=> SOFTMAX layer. Two of the CONV layers will acquire knowledge of thirty-two 3 × 3 cores. Two more CONV levels, which have 64 cores each, will also be 3 × 3. With 2×2 cores and a 2×2 stride, the POOL layer will execute a Maxpooling operation.

Chapter – 4

RESULTS AND DATA INTERPRETATION

4.1 INTRODUCTION

It is essential to have a goal when conducting research in any field or subject. Research is conducted throughout most professions; more than a set of abilities research is a method of thinking, analyzing critically the numerous parts of our day to day professional job, understanding & establishing guiding principles that control a certain procedure.

Research is a systematic process of gathering and analyzing data to deepen our comprehension of the topic at hand. After being gathered, data must be methodically analyzed in order to produce the desired results and their scientific interpretations. Consequently, it is critical that the data be supplied in a properly organised manner in order for it to explain the study's purpose.

The analysis & interpretation of the data are two elements yet they cannot be separated since if we separate them, the remaining one has no relevance. Actually, the research endeavour has used the approach of analysis and interpretation from the very beginning. In addition, the data analysis & interpretation are true representations of the work. In light of the study's aims, it is evident that machine learning IDSs that are currently available do not meet the standards for a variety of reasons; analysis describes the data- produced results, and interpretation explains the relevance of those results. In order to aid network IDS designers, this study employs DL approaches like RNN, Deep-NN, & design algorithms.

4.1.1 DL Algorithms

Modelling a deep network is made possible by DL, a variant of ML that involves removing multiple layers. Due to their structural depth and simplicity of self-learning, these approaches output more relevant features from a dataset, allowing them to beat ML in terms of throughput. This study employs DL techniques such as

RNN & Deep- NN, together with algorithm design, to help the creation of network IDS.

4.1.2 Recurrent Neural Networks

RNNs expand upon the strengths of regular feed-forward NN to better model sequence data. The three main components of a RNN are the input unit, the hidden unit, & two output units, which double as memory. This conclusion is based on the most recent input & output of each RNN unit. RNNs are widely used in many fields, including semantic comprehension, handwriting prediction, audio processing, & activity detection. Recurrent Neural Networks (RNNs) are able to extract features and analyse IDSs. Random neural networks (RNNs) struggle with short-term memory since they can only handle sequences up to a particular length. Some have proposed L STM & GRU variants of RNNs as answers to these issues. Yin et al. demonstrated an RNN- based IDS capable of both binary & multi categorization. Experimental runs with varying hidden node densities & training rates allowed us to gauge the model's robustness. The effectiveness of the model is influenced by the learning rate & quantity of hidden nodes. The best results were achieved in binary and multiclass settings. Due to the longer time needed to train the model for the R2L & U2R classes, detection rates have been reduced. But, other DL methods that are presently being used in this research have no comparison to the proposed strategy.

4.1.3 Autoencoder

AE is a famous unsupervised NN & deep learning technique. During the learning process, its goal is to provide an output that closely resembles the input as much as possible. Despite having same sized input & output layers, concealed layers are typically smaller. The encoder-decoder function in AE allows for asymmetric encoding & decoding. There are several varieties of AE, including stack, sparse, & variational AE. An IDS based on radio frequency identification (RFID) & utilising deep AE and ML was developed by Shone et al. The model can be made computational & temporally effective by using only the encoder portion of AE to achieve nonsymmetric functioning. Both asymmetric deep-AE stacks contain three hidden layers. It was RF who did the classifying. The present state-of-the-art Deep

Belief Network is surpassed by the proposed approach, which offers superior detection accuracy with reduced training time (DBN). The algorithm was unable to detect R2L & U2R assaults due to insufficient training data. Yan et al.67 suggested incorporating SSAE & SVM into an IDS. Data would be categorised with SVM after features were extracted using SSAE. While conducting tests, we take into consideration the difficulty of both binary and multi categorization. Its proposed model outperformed alternatives that made use of feature selection, ML & DL. The model's ability to identify U2R & R2L attacks is commendable when related to other types of attacks in the dataset. An analogous method of self-teaching was created by A-Qatf et al. utilising sparse AE and support vector machines. They put the suggested model through its paces on the NSL-KDD dataset to gauge its efficacy. Overall, the outcomes presented that the DL & ML model performed better than the competition. There has been no investigation into how well the proposed approach works in the R2L and U2R classes.

4.1.4 Deep Neural Network

The essential DL structure, DNN, permits the model to learn several layers. Layers for input, output, and extra information are included in that. Modelling complex, nonlinear functions is now within reach, thanks to DNN. Adding more hidden layers to the model makes it work better. One fully connected layer's output layer employs a softmax classifier for categorization purposes. Results proved the proposed model was reliable even without U2R attack data. More sophisticated structures, with more nodes and layers, require more time and resources for calculations, based to the study's authors. To resolve these issues, an optimisation method with integrated automated adjustment is necessary. Wang et al investigated thoroughly to find out how each factor aided in the escalation of aggressive incidents. Attacks such as FGSM, JSMA, DeepFool, and CW produced negative examples. If the network is to be protected against attacks, it is imperative that the most frequently utilised attributes receive additional attention.

4.1.5 Convolutional Neural Network

When dealing with array-structured data, CNNs perform admirably. It uses a partially connected layer, a softmax classifier, a stack of convolution and pooling layers, an

input layer, and so on to facilitate feature extraction & classification. Compared to previous computer vision methods, CNN is light years ahead. In order to classify & extract features, the IDS system makes use of them. By utilising convolutional neural networks, Xiao et al. presented an efficient IDS. PCA & AE are used to extract unique properties first. A CNN cannot be used without first transforming the feature set into a 2- dimensional matrix. In these tests, the KDD Cup'99 dataset was utilised. Algorithm training & testing can be done more quickly with this approach. The detection rates for R2L & U2R attacks are lower than other types of attacks. CNNs & gradient-based choice forests (gcForests) form the basis of the multilayer IDS model developed by Zhang et al. For the purpose of transforming raw data into 2-dimensional grayscale images, they also suggested a novel P-Zigzag approach. Initially, detection was carried out using a coarse grit layer derived from a superior CNN model (GoogLeNetNP). In the fine-grained layer, the abnormal classes are subdivided into N-1 classes (caXGBoost) using gcForest. In this study, data sets from the UNSW-NB15 & CIC- IDS2017 initiatives were combined. While the individual strategies increase detection rate & accuracy, the proposed model decreases false alarm rate. As a strong IDS, Jiang et al. suggested a configuration of hierarchical CNN with bidirectional long-term memory (BiL-STM). In order for the model to learn all properties, it is essential to amplify minority samples utilising the SMOTE technique. We used BiLTSM to get the time data and CNN to get the spatial features. Research datasets used in the study were NSL-kDD & UNSWNB15. The proposed method considerably improves accuracy and detection rates. The dangerously low detection rate for assaults on minority data classes is in comparison to other forms of attacks, although it has grown slightly. Because of the building's complexity, the training duration is lengthier than usual. Our IDS model, FSL, was built by incorporating the innovative concept of Few-shot Learning (FSL) into the existing DL architecture. There would be minimal utilisation of the labelled data in the training process. Embedding techniques, such as DNN and CNN, enable feature reduction & extraction of the most relevant features. The model achieved decent detection rates in differentiating minority attack classes, as demonstrated in studies on the UNSW-NB15 & NSL-KDD datasets. The proposed model was trained with just 2% of the data, however it still managed to reach this remarkable result.

4.2 EXPERIMENT 1

In this experiment, numerous important ML & DL methods are used, such as RF, SVM, XGBoost, LSTM, AlexNet (Mini-VGGNet), & endlessly more. Compared to other oversampling techniques like RUS, ROS, & SMOTE, its 30 methods are more extensive. The idea to implement intrusion detection was actually ours. The initial stage of our IDS is data pre-processing. Issues like detecting outliers, filling in missing values, & removing duplicates are all part of this process. Before we can process the training set, we partition the data into a training set & test set using a DSSTE data balancing approach. Prior to beginning the modelling process, we utilise Standard Scaler to standardise the data & digitise the sample labels. Convergence is helped by this. Utilising the training data, the classification model is trained, and then it is tested with the test data.

Propose 1

Step 1:

Collect the dataset, this dataset contains intrusion website information.

Step 2:

Performing EDA on the dataset and get to know that it can be done as binary classification and multi-class classification.

Step 3: Processing

- Dropping Null values.
- Removing duplicate Values
- Changing To scalar values
- Feature Extraction

Step 4:

Plotting graphs and done final processing on the data for the training.

Step 5:

Creating an AutoEncoder Deep Learning model & fitting the data to it, let it train. After completion, utilize the model for testing.

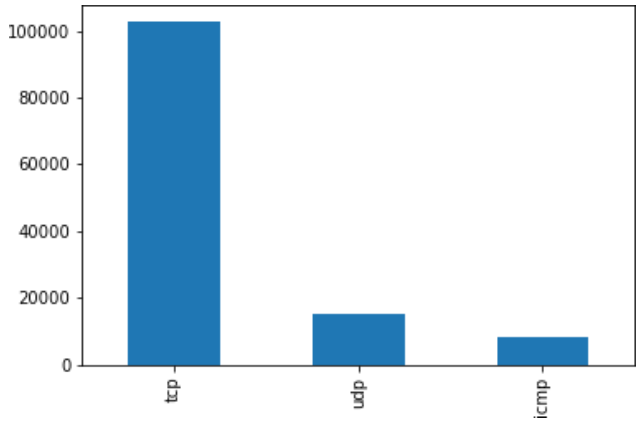
Step 6:

Evaluation of the model, testing the model on the test set & measuring the performance in terms of precision, recall & F1-Score. The AutoEncoder Deep learning model performed very well.

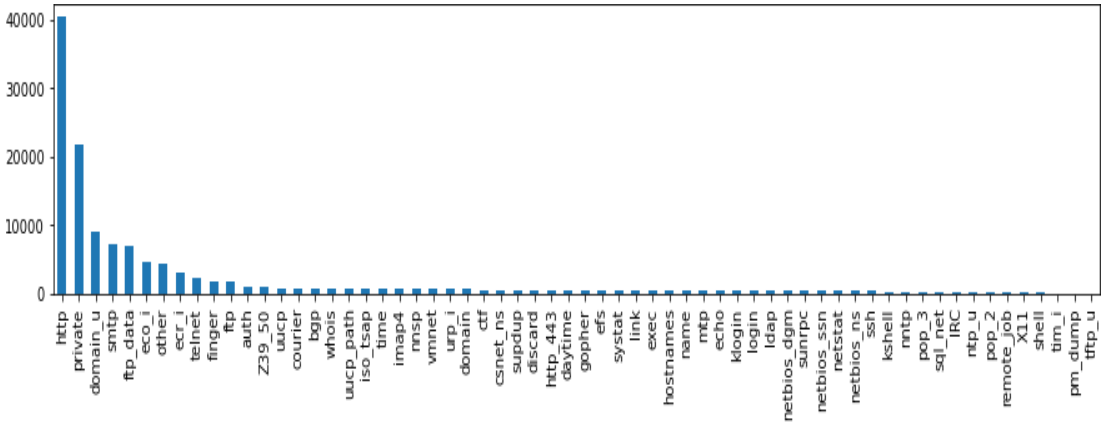
METHODOLOGY

Steps used in coding Pre-Processing

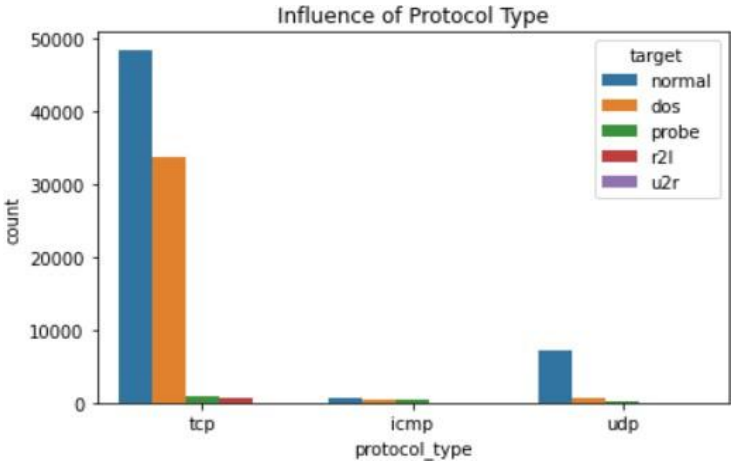
1. Bar Graph protocol_type



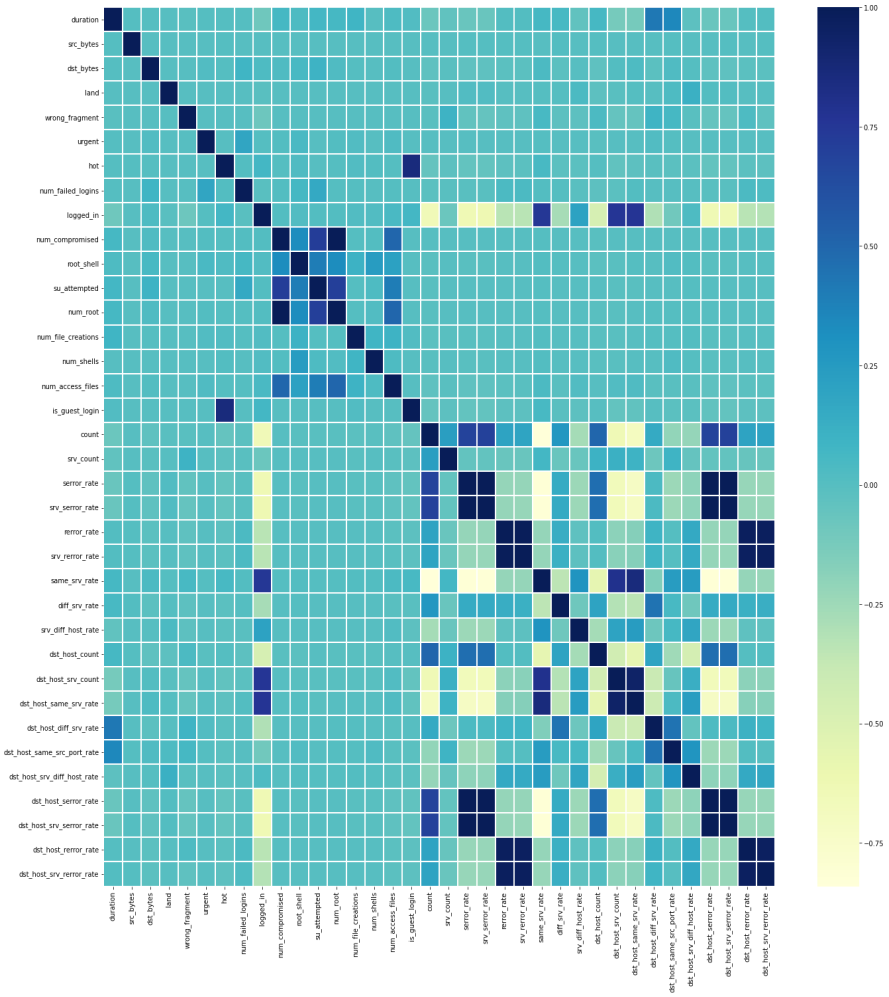
2. Every Sevrcie Graph



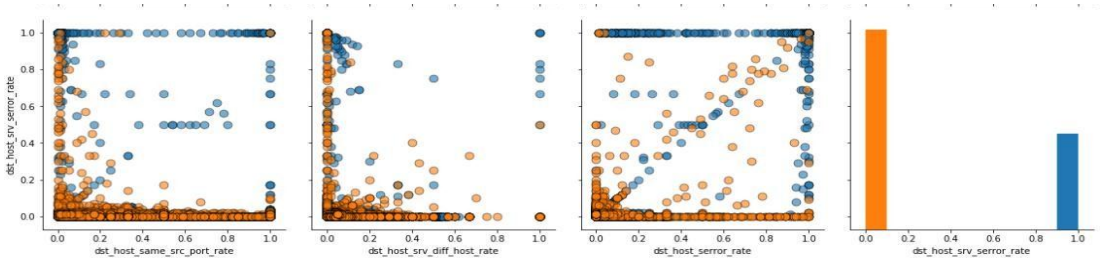
3. Protocol type influence on target



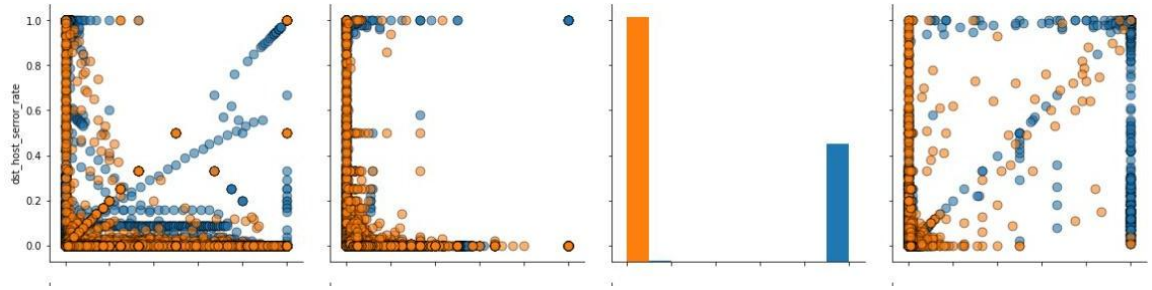
4. Correlation between whole data



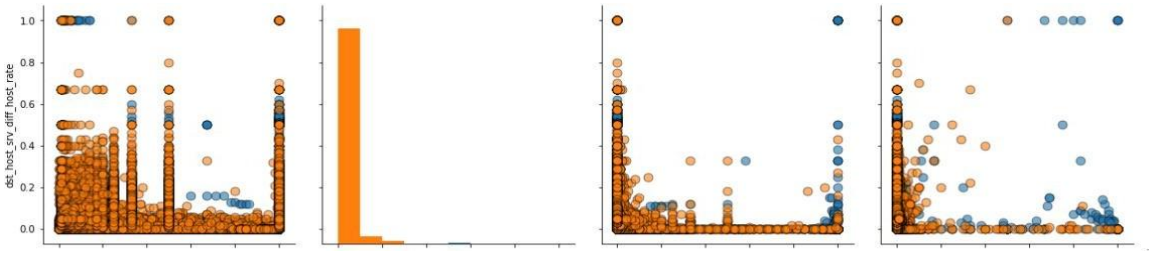
5. Dst_host_port



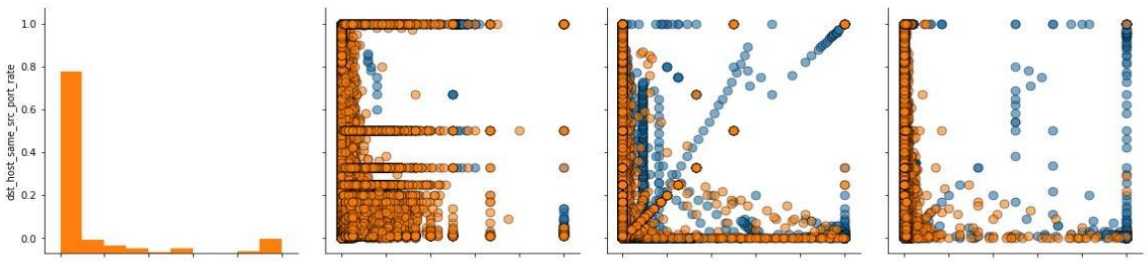
6. dst_host_serror_rate



7. dst_host_srv_diff_host_rate



8. dst_host_same_src_port_rate



9. **Pre-processing:**

- Dropping Null values.
- Removing duplicate Values

- Changing To scalar values
- Feature Extraction

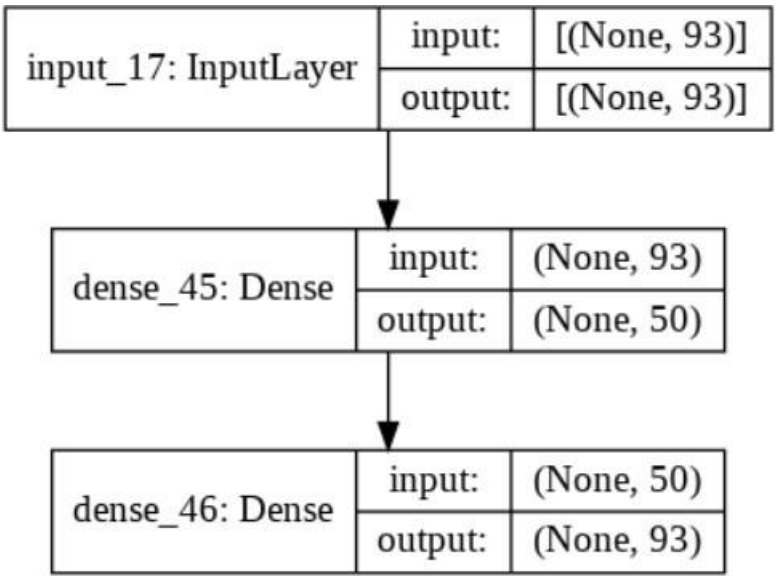
10. Model Summary

a. Optimization Function = adam

b. Loss function = mean_squared_error

c. Encoding dim = 50

```
Model: "model_19"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_19 (InputLayer)        [(None, 93)]              0
_________________________________________________________________
dense_51 (Dense)             (None, 50)                4700
_________________________________________________________________
dense_52 (Dense)             (None, 93)                4743
=================================================================
Total params: 9,443
Trainable params: 9,443
Non-trainable params: 0
_________________________________________________________________
```



Evaluation Metrics

Accuracy, Prediction, Recall, & F1-Score are utilized to estimate the effectiveness of the experimentation model. These evaluation criteria take into account the intrusion detection system's false alert rate and flow recognition accuracy rate. Depending on how well the model predicts the actual label, it can be divided into 4 different groups. A positive sample that is mistakenly construed as a negative sample is known as a false negative (FN). False Positive (FP), a sample that was supposed to be negative but was actually read as positive. A sample is properly referred to as being truly negative (TN) when it is. FP samples, or those that are not actually positive, are regarded as false positives. These measurements are calculated using equations.

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

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

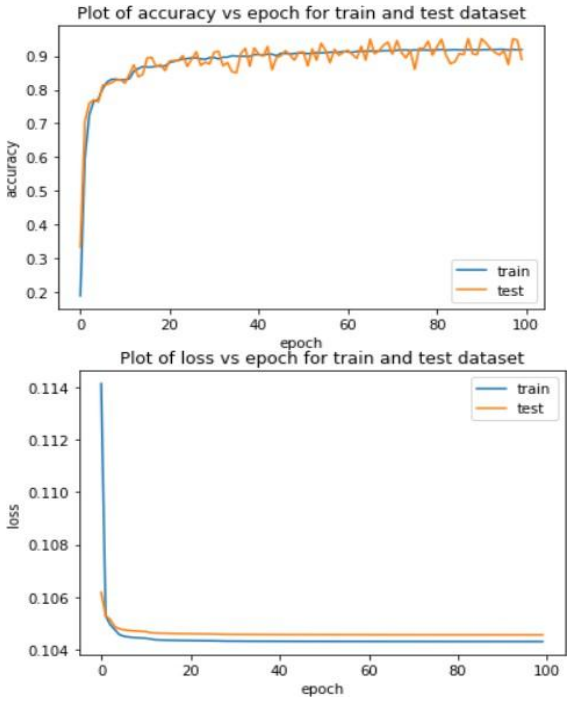
$$Recall = \frac{TP + TN}{TP + TN + FP + FN}$$

$$F1\_Score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

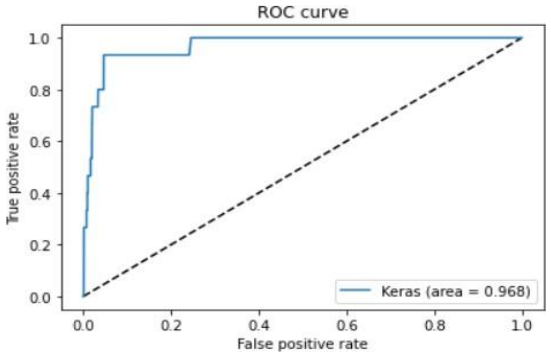
Evaluation Metrics

Model	Accuracy	Precision	Recall	F1-Score
Base Alex	0.82	0.83	0.82	0.81
Base LSTM	0.78	0.78	0.78	0.75
Base XG Boost	0.77	0.81	0.77	0.73
Proposed	0.88	0.86	0.88	0.90

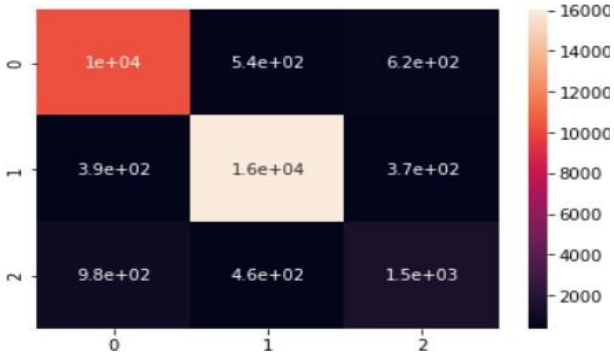
Accuracy, loss & Result Graph:



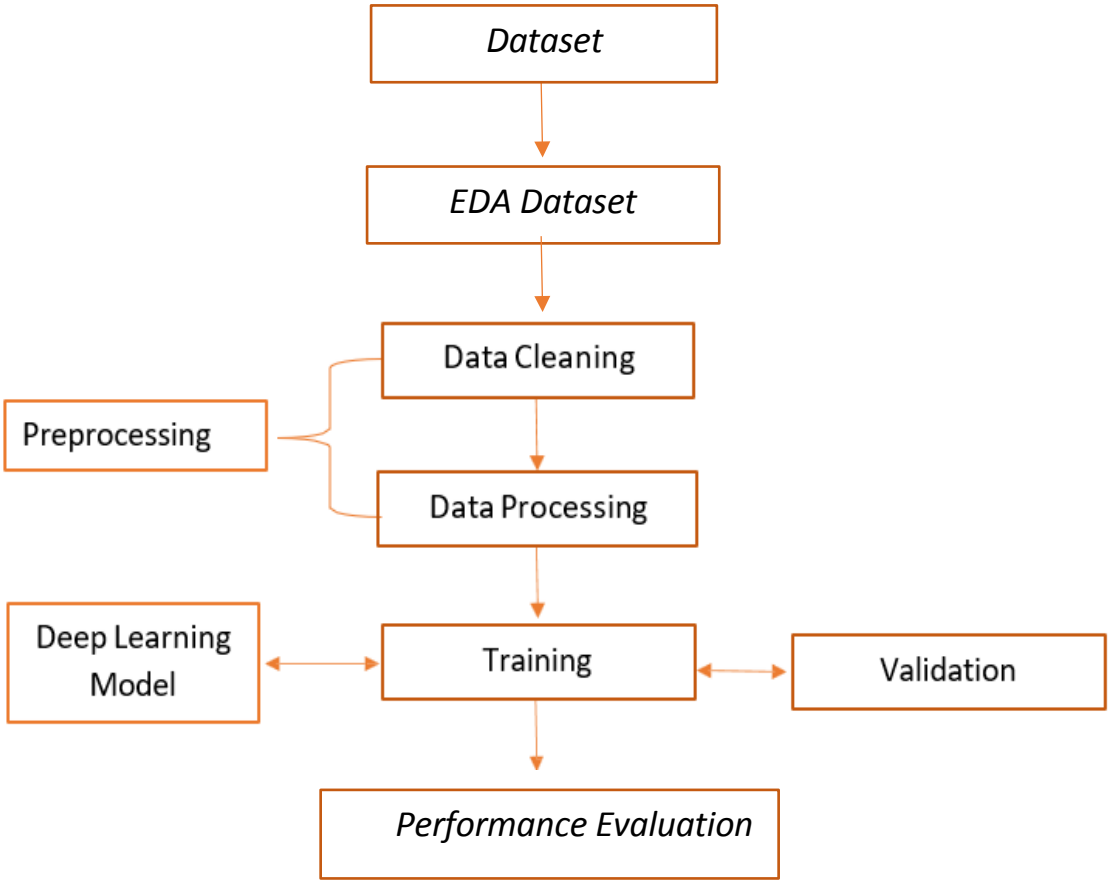
ROC Curve:



Confusion Matrix:



Flow Chart:



4.3 EXPERIMENT 2

A first test of the classifier's efficacy was conducted on a deflated version of the training set. In the described DSSTE approach, K is a parameter scaling factor. The number of problematic samples increases as K increases within a certain range, but stays constant outside of that range. However, adjusting K will improve majority compression & minority augmentation when dealing with tough samples. As a precaution, we performed this to make sure the data sample was related, didn't produce too much noise, and that our DSSTE method could get the best possible results from the sample.

In NSL-KDD & CSE-CIC-IDS2018, we utilized a number of different scaling factors K to control the training data. Six proposed classifiers were put to the test, with the average F1-Score of the six used to determine which performed the best.

When trained with K D50, NSL-KDD classifiers perform exceptionally well, on average. In the CSE-CIC-IDS2018, a classifier with a mean score of K D 10 is considered to be above average. As a result, we scaled everything by a factor of k D 50 based on the typical F1-Score for NSL-KDD, and we reduced the size of the problematic samples in Normal/DoS/Probe while improving them in R2L/U2R. For instance, the CSE-2018-CIC-IDS2018 utilized the scaling factor K D 10 & NSL algorithm to deal with the challenging samples in NSL-KDD. A new exercise set was among the first things I did after the operation. We outperform the state-of-the-art sampling approaches with our proposed DSSTE algorithm on both the NSL-KDD and CSE-CIC-IDS2018 datasets.

Based on the results demonstrated in the NSL-KDD dataset, LSTM trained on the original training set has the potential to attain a maximum accuracy of 78% and an F1- Score of 75%. It was found that miniVGGNet had the best accuracy rate and XGBoost had the best recall rate after sampling the RUS algorithm's training. For instance, the LSTM was able to attain a 75% recall rate by sampling from the training data of the ROS algorithm. AleNet's 82% accuracy and 82% recall after SMOTE sampling the training set was best of all models. Here we report that AleNet, using data from a training set sampled by DSSTE, achieves an accuracy of 82% & recall of 81%.

In terms of accuracy (88%) & F1-Score (90%) on the CSE-CIC-IDS2018 dataset's raw training set, RF is the clear winner. Following the RUS, ROS, & SMOTE sample protocols as outlined in the training. The RF was the optimal choice when taking accuracy and F1-Score into account. Therefore, the efficiency benefits were negligible at best. When tested on a training set sampled using the DSSTE method described here, MiniVGGNet obtained the best results in relations of accuracy & recall. As an outcome, random forest's recall & accuracy are on par with those of competing approaches. Random forest shows the generalizability of integrated learning with each sample method while utilising less hardware resources. For each test instance, we determined the classifier's mean accuracy and F1-Score. Compared to the initial sampling method, RUS, ROS, & SMOTE make better on the LSTM dataset. The F1-score and forecast accuracy have both made very little headway.

Since the initial version of the approach was introduced, there has been an approximately 8% improvement in accuracy and a 7% improvement in F1-Score for DSSTE. Notably, the CSE-CIC-2018 dataset does not show any gain in performance when utilizing the RUS, ROS, or SMOTE sampling methods. Once the training set is finished, the DSSTE sampling method utilised in the present investigation improves average F1-Score by 1% and average accuracy by 2%.

An effective measure of a classification model's performance, the F1-Score is obtained by combining the prediction and recall rates. For this reason, we compare the methodologies put forward by various authors using the F1-Score & accuracy. We demonstrate that, on KDD Test C, our proposed data sampling technique, DSSTE, performs better than prior techniques. Although building a model takes time, the F1- Score is relatively similar to that of AESMOTE, demonstrating that reinforcement learning is beneficial for automatic paired sequence learning. Because of this, our suggested approach shines in networks where traffic is not evenly distributed.

The enormous and redundant CIC-IDS-2018 dataset has been picked and studied in various ways by various academics. Hence, we did not compare our results to those of other studies that utilised the CIC-IDS-2018 dataset. When compared to other sample approaches, our research shown that the DSSTE method worked far better. The CIC- IDS-2018 dataset is a great test bed for DSSTECAlexNet. Detection rates for brute force & infiltration attacks are now near 100% in several instances.

Although typical sampling methods might be useful for balancing a training set, keep in mind that they cannot provide you a distribution that perfectly matches the actual data. Using the ROS approach instead of the RUS strategy can lead to data redundancy & overfitting. However, because SMOTE interpolation tends to increase noise traffic and data overlap, the training set becomes more difficult. Fixing training set imbalances through augmenting & condensing problematic data is a key component of our proposed DSSTE technique. By incorporating additional data, the classifier is able to produce more precise outcomes.

Propose 2

Step 1:

Collect the dataset, this dataset contains intrusion website information.

Step 2:

Performing EDA on the dataset and get to know that it can be done as binary classification and multi-class classification.

Step 3: Processing

- Dropping Null values.
- Removing duplicate Values
- Changing To scalar values
- One Hot Encoding
- Changing Text labels to number
- Data Normalization
- Feature Extraction

Step 4:

Plotting graphs and done final processing on the data for the training.

Step 5:

Creating a Conv1D Deep Learning model and fitting the data to it, let it train. After completion, use the model for testing.

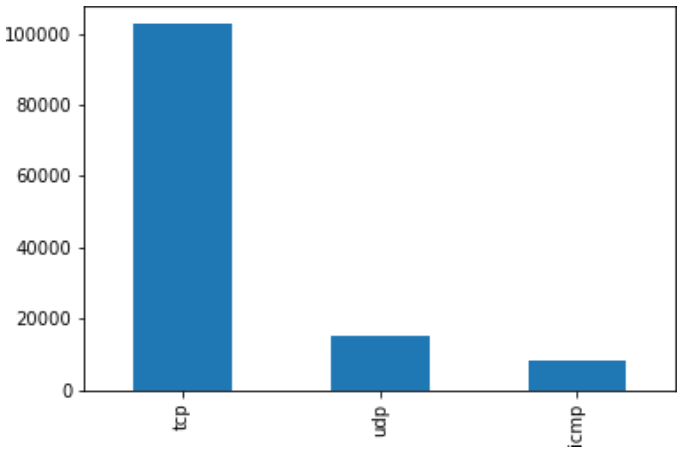
Step 6:

Evaluation of the model, testing the model on the test set and measuring the performance in terms of precision, recall and F1-Score. The Conv1D Deep learning model performed very well.

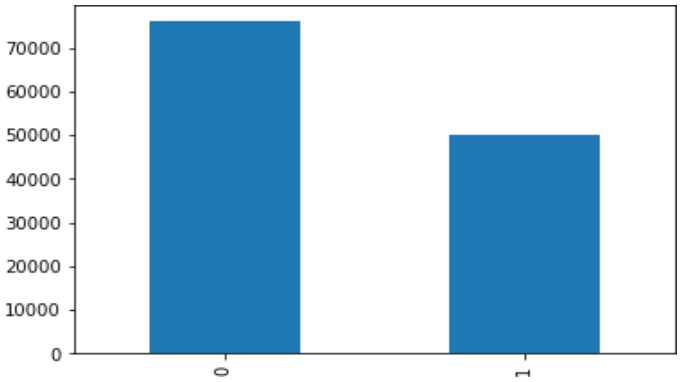
METHODOLOGY

Steps used in coding **Pre-Processing**

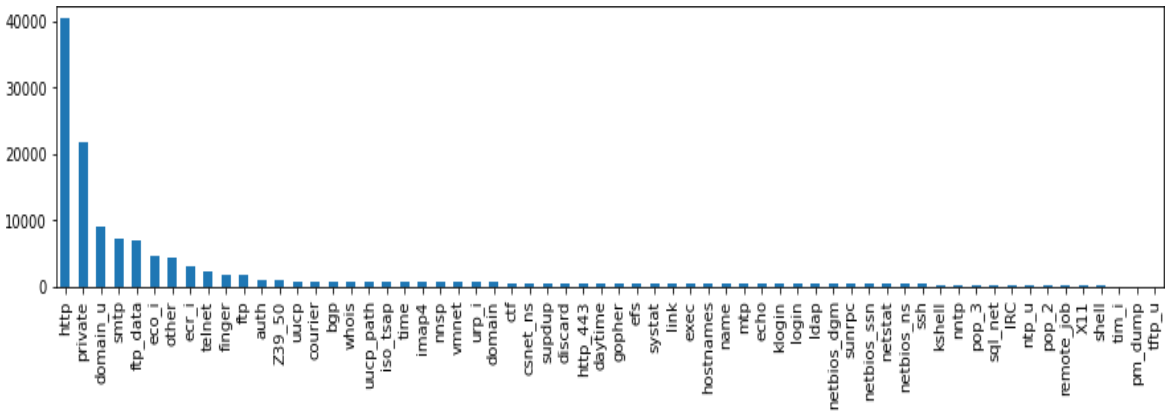
11. Bar Graph protocol_type



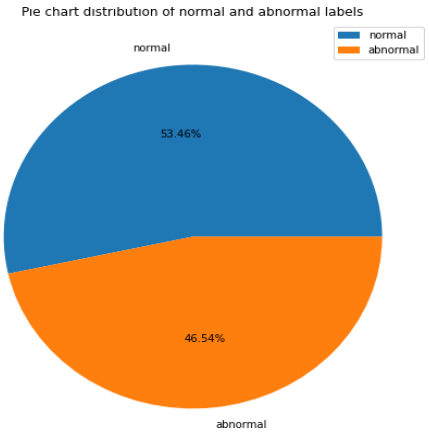
12. Log in Bar Graph



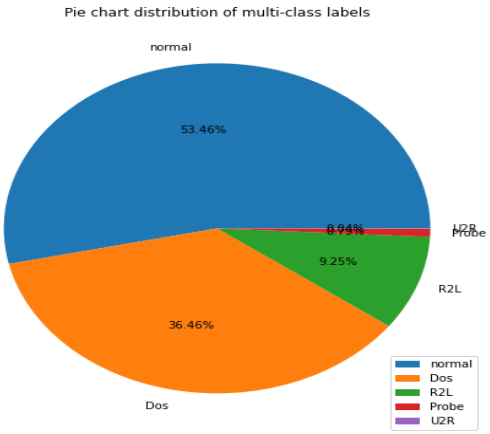
13. Every Sevrcie Graph



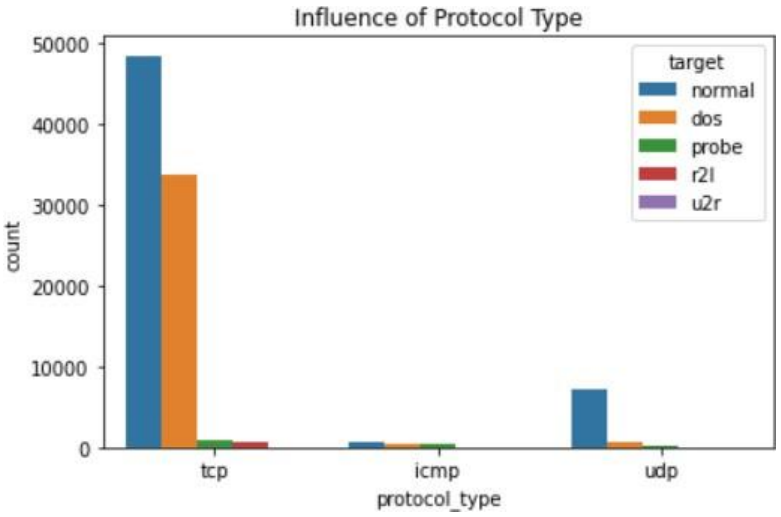
14. Pie Chart distribution of binary class



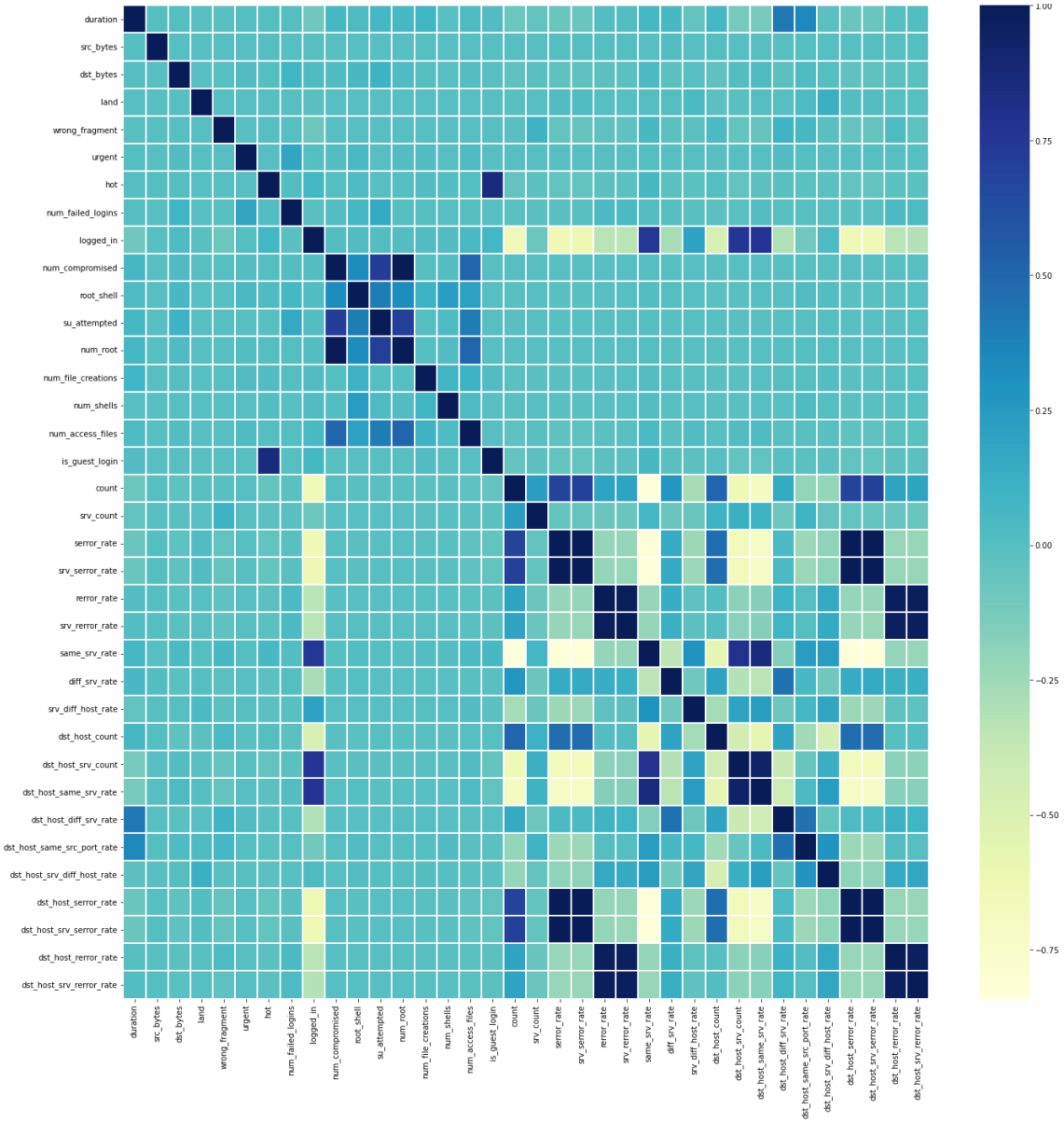
15. Pie Chart distribution of multi class



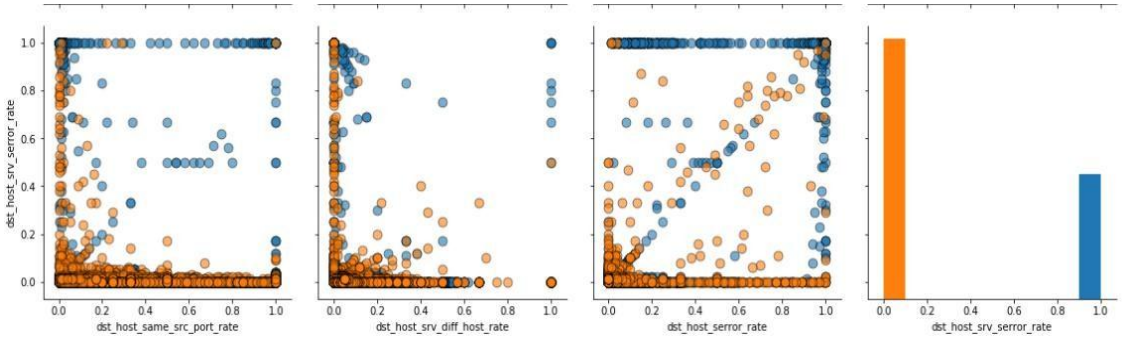
16. Protocol type influence on target



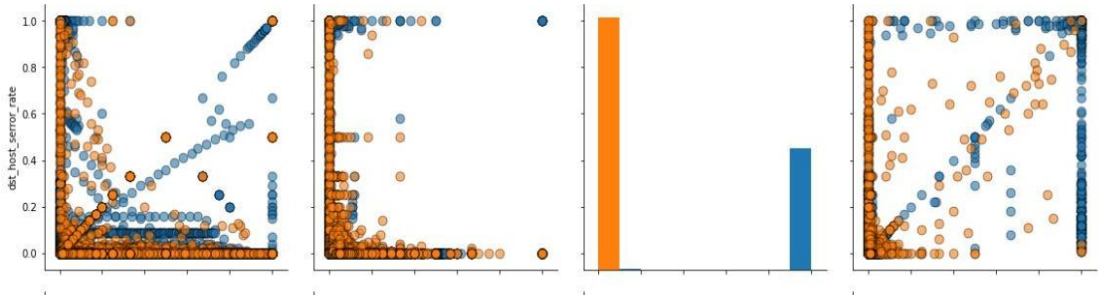
17. Correlation between whole data



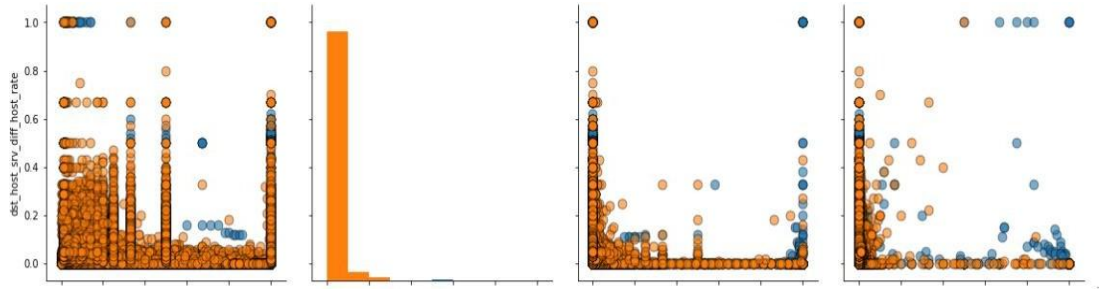
18. Dst_host_port



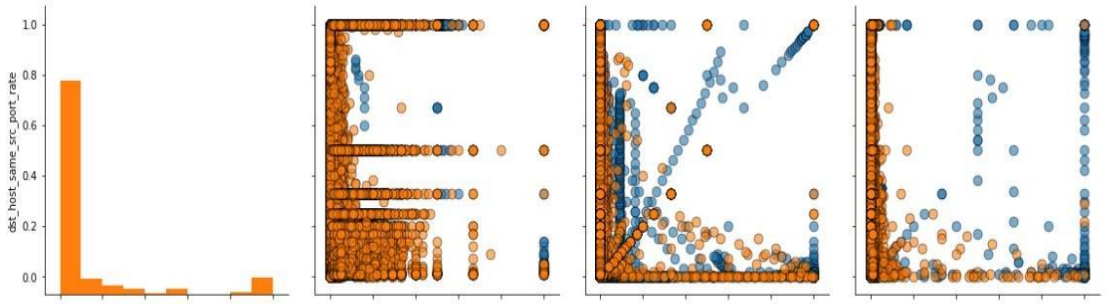
19. dst_host_serror_rate



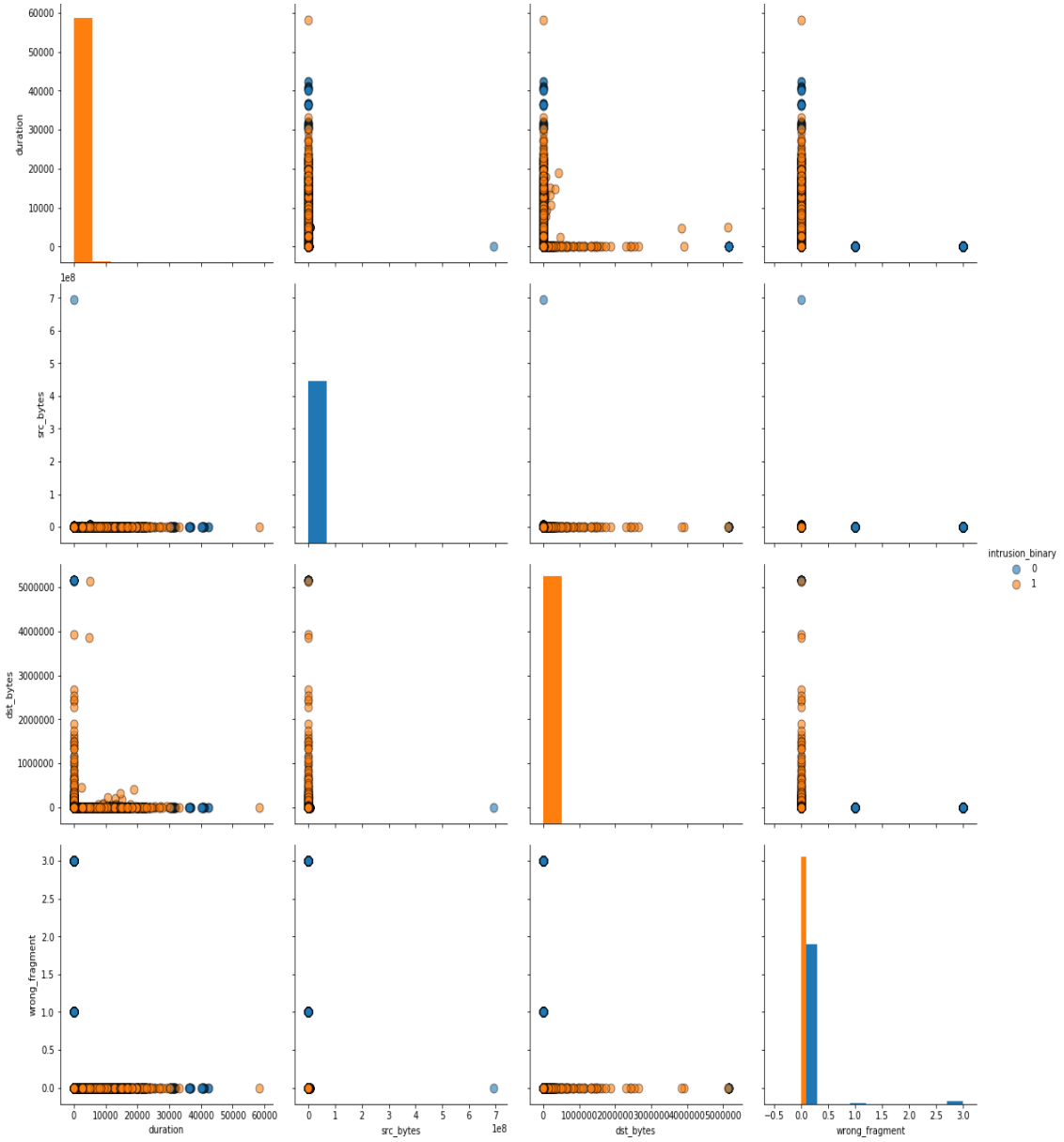
20. dst_host_srv_diff_host_rate



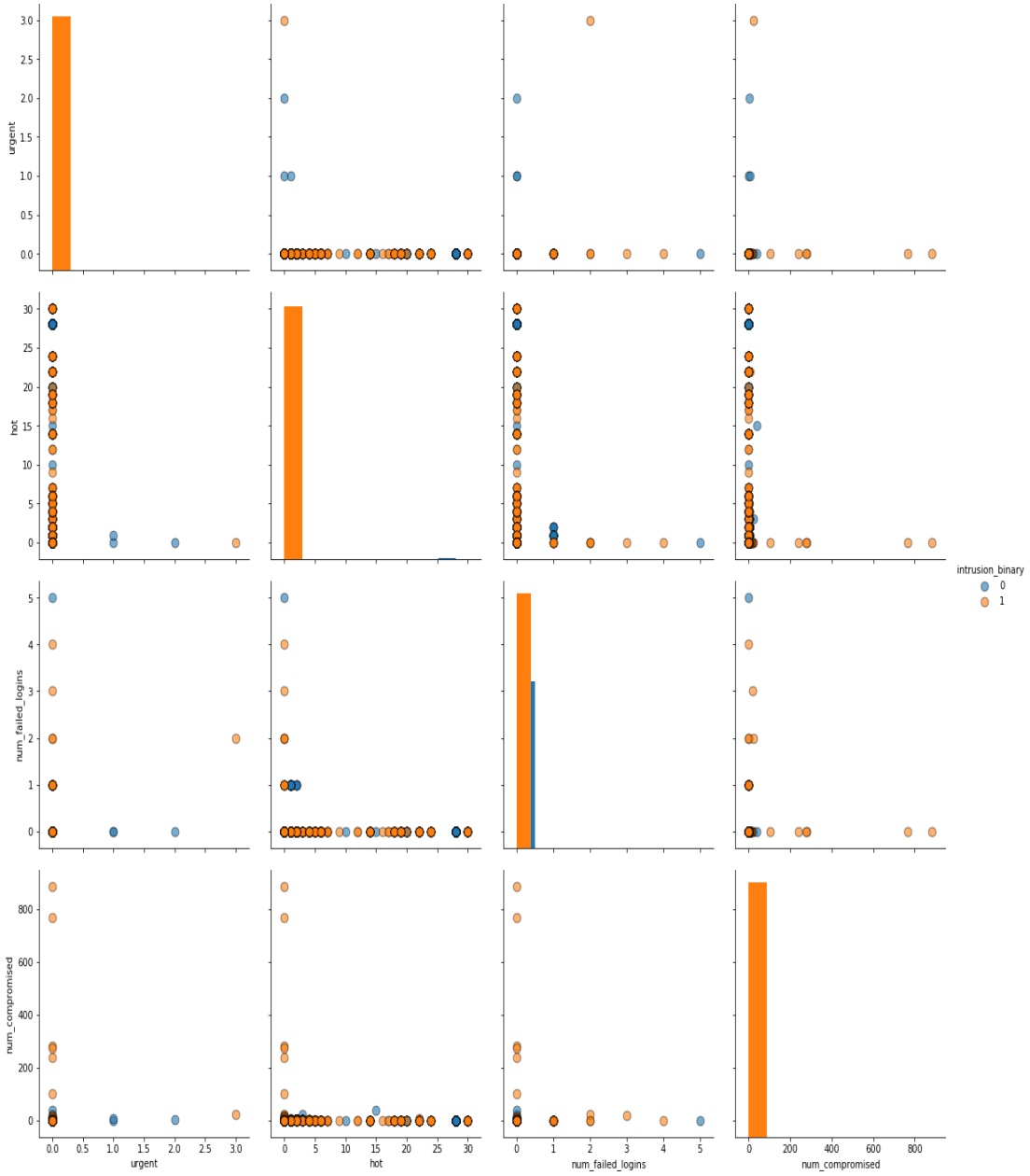
21. dst_host_same_src_port_rate



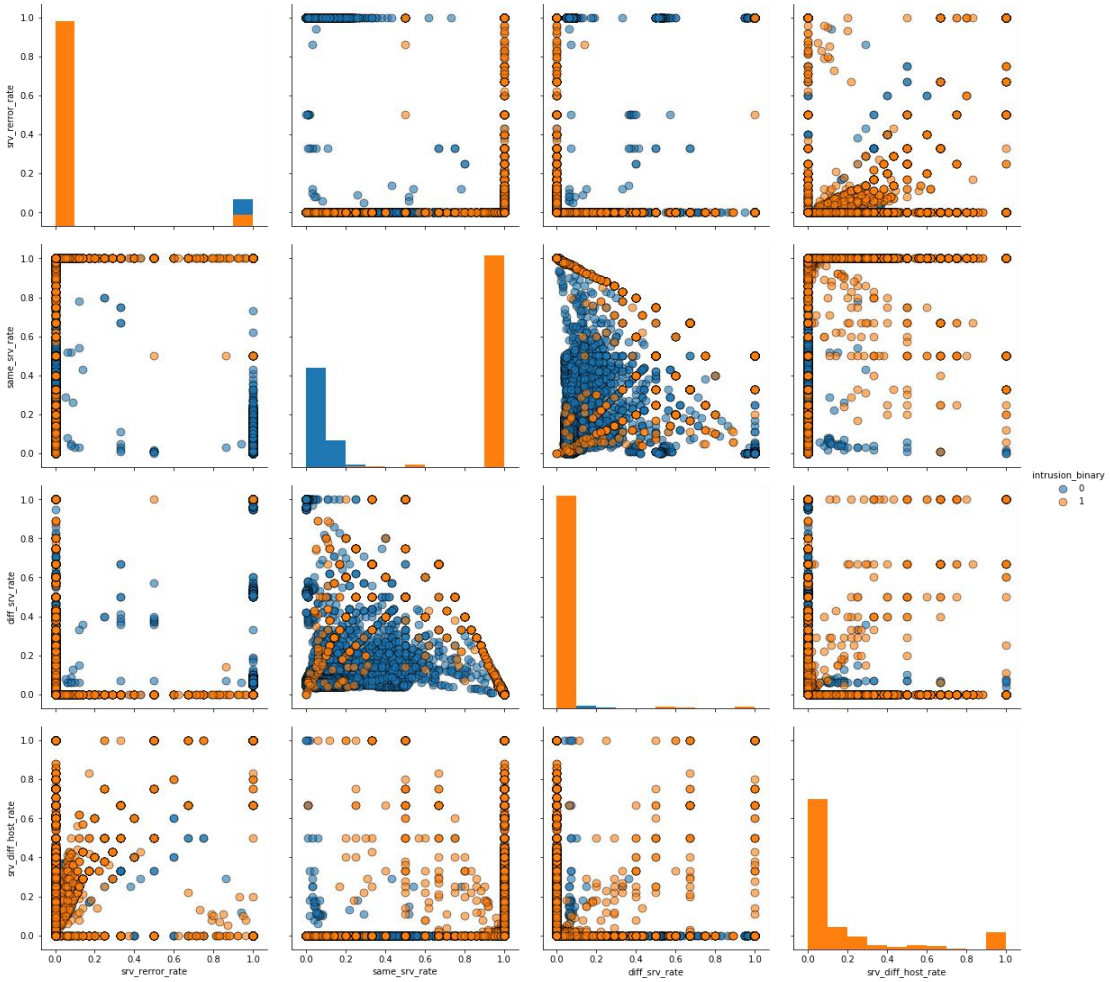
22. features=['duration', 'src_bytes', 'dst_bytes', 'wrong_fragment']



23. features=['urgent','hot','num_failed_logins','num_compromised']



24. features=['srv_rerror_rate','same_srv_rate','diff_srv_rate','srv_diff_host_rate']



25. Pre-processing:

- Dropping Null values.
- Removing duplicate Values
- Changing To scalar values
- One Hot Encoding
- Changing Text labels to number
- Data Normalization
- Feature Extraction

26. Model Summary

a. Conv1d(32,3,padding=same,activation=relu)

b. MaxPool(pool_size=4)

c. Dense(16,activation=relu)

d. Dense(5,activation=softmax)

e. Loss = categorical_crossentropy

f. Optimizer = adam

```
Model: "sequential_6"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d_4 (Conv1D)            (None, 93, 32)            128
_________________________________________________________________
max_pooling1d_2 (MaxPooling1 (None, 23, 32)            0
_________________________________________________________________
layer_normalization_2 (Layer (None, 23, 32)            64
_________________________________________________________________
flatten_2 (Flatten)          (None, 736)               0
_________________________________________________________________
dense_10 (Dense)             (None, 16)                11792
_________________________________________________________________
dropout_2 (Dropout)          (None, 16)                0
_________________________________________________________________
dense_11 (Dense)             (None, 5)                 85
=================================================================
Total params: 12,069
Trainable params: 12,069
Non-trainable params: 0
_________________________________________________________________
```

Evaluation Metrics

The detection module's measuring system is the primary use case for this framework. To measure its performance, four metrics can be used: accuracy, recall, preference, and overall performance. When evaluating the efficacy of IDS, several metrics are often used. Their formulations are as follows.

The proportion of correctly identified samples relative to the total number of samples is termed accuracy (Acc), and it is a measure of the overall performance of a model

$$\mathrm{Acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}.$$

Recall, also known as detection rate (DR), can be evaluated in two ways: first, by totalling the samples that were appropriately categorised into a specific class, & second, by totalling the samples that really belonged to that class.

$$\mathrm{Re} = \mathrm{DR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}.$$

The precision of a classification system can be determined by dividing the totalling the samples accurately classified into a category by the totalling the samples in that group.

$$\mathrm{Pr} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}.$$

The precision & recall are calculated by averaging the F1-score (F1) or F-measure harmonically.

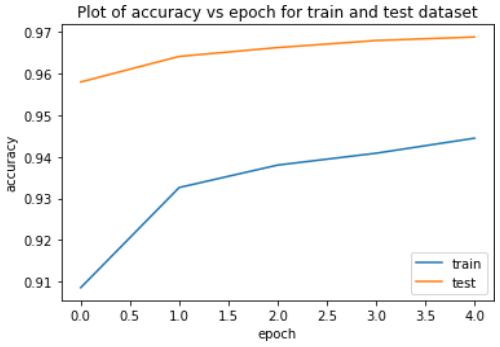
$$F1 = \mathrm{FM} = \frac{2 \cdot \mathrm{Pr} \cdot \mathrm{Re}}{\mathrm{Pr} + \mathrm{Re}}.$$

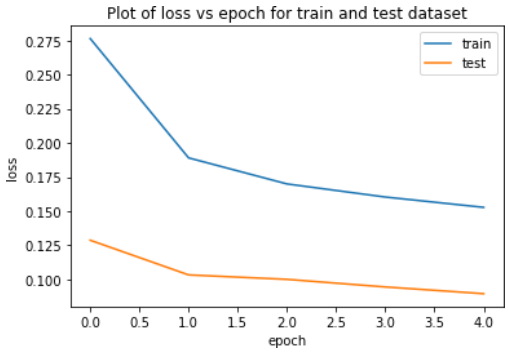
With respect to each sample type, the TP value indicates the quantity of samples that were appropriately classified as that type, the TN value the quantity of samples that were appropriately classified as not that type, the FP value the quantity of samples that were misclassified, & FN value the quantity of samples that were mistakenly classified as that type.

Evaluation criteria: Proposed framework comparison

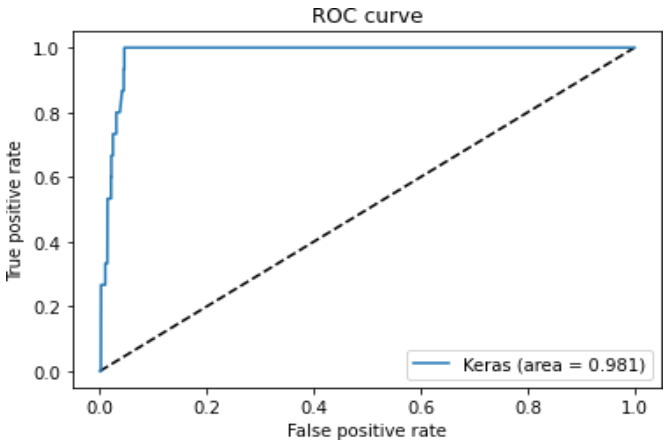
Model	Accuracy	Precision	Recall	F1-Score
Base	0.82	0.83	0.82	0.81
Base LSTM	0.78	0.78	0.78	0.75
Base XG Boost	0.77	0.81	0.77	0.73
Proposed 1	0.88	0.86	0.88	0.90
Proposed 2	0.96	0.96	0.98	0.97

Accuracy, loss & Result Graph:

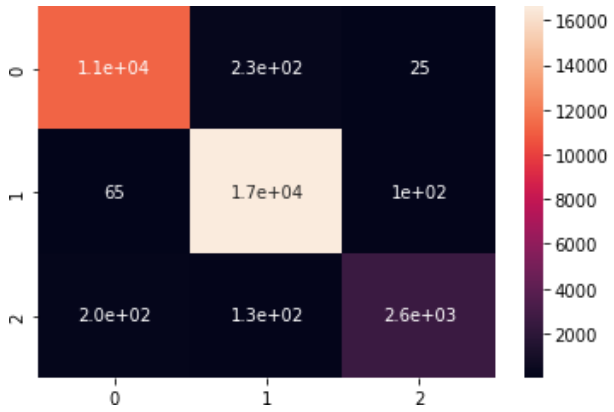




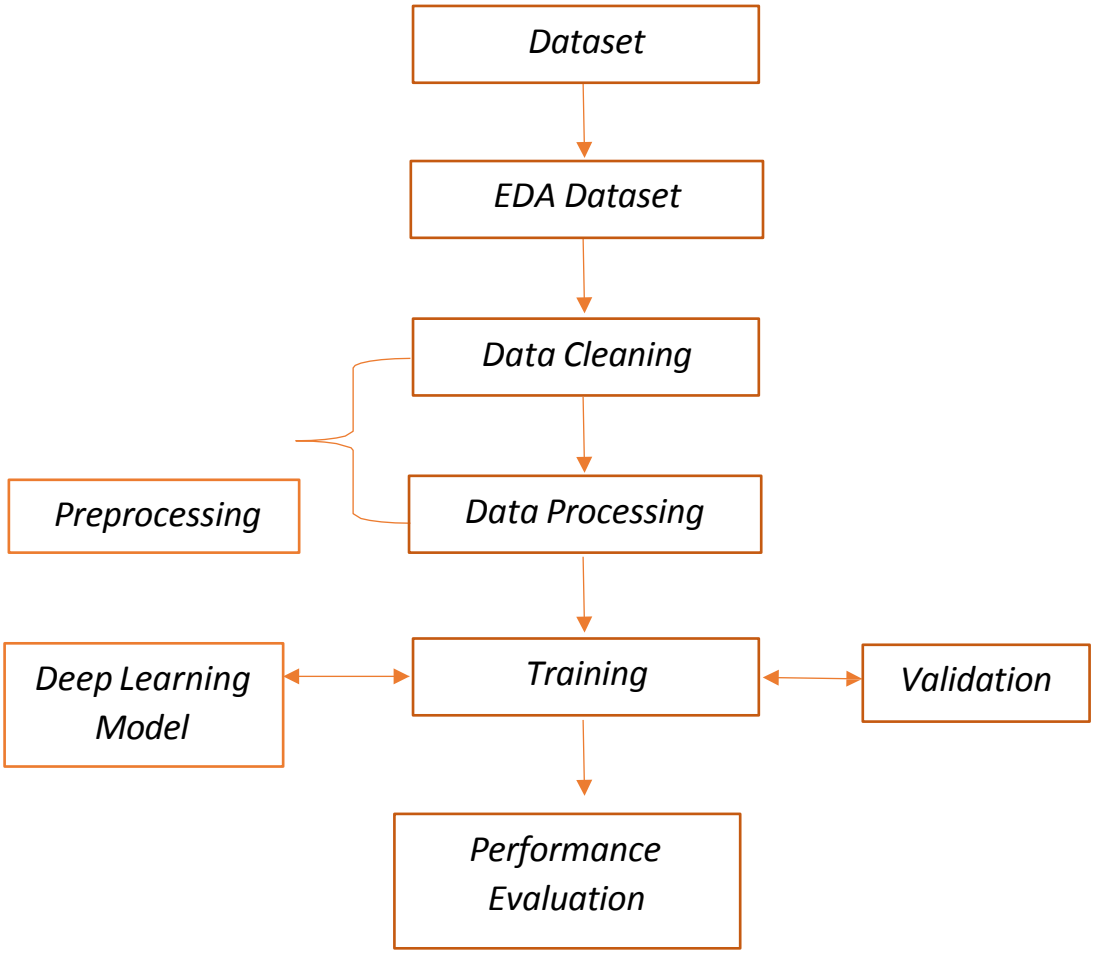
ROC Curve:



Confusion Matrix



Flow Chart:



Chapter – 5

CONCLUSION AND FUTURE SCOPE

5.1 CONCLUSION

The importance of network security research has increased over the past ten years as a outcome of the growing fascination with the advancement of internet and communication technologies. Technologies like firewalls, antivirus software, & IDS are used to protect the network & associated assets in cyberspace. NIDS, one of these systems, continuously scans network traffic for questionable or hostile activities. In 1980, Jim Anderson developed the idea for IDS. In order to fulfill the rising need for network security, a excess of IDS have been established since then. But over the past ten years, the size of networks & variety of applications managed by network nodes have both exploded the result of technological developments. As an effect, a huge amount of useful information is produced & spread throughout the network. Protecting this data & network nodes has become a difficult challenge as a result of the development and growth of a significant number of new assaults. A network's nodes are susceptible to hacking at some time. The data node of a company, for instance, may be crucial.

With the widespread usage of computers & networks, destructive activity has increased in the complicated framework of contemporary civilization. Big data, IoT, & cloud computing are just a few of the new technologies that are fueling this expansion. IDS is becoming more and more crucial as the requirement to defend network resources from cyber-attacks increases. We can detect, name, & recognize abnormal behavior in networks and computer systems brought on by hackers thanks to IDS. ML methods, for instance, can be employed to predict & classify features beforehand. Depending on the classifier's training technique, IDS can be broken down into the supervised & unsupervised learning methods of learning from labeled & unlabeled training examples, respectively. Among other methods, researchers have employed ANN, RF, SVM, KNN, and others to find intrusions. Researchers

intrigued by the application of DNNs in intrusion detection have given CNN a lot of thought.

The primary focus of this research will be on feature extraction from datasets using AE model. Unsupervised DNNs can learn features from unlabeled input, unlike supervised DNNs. An integral part of AE is the encoder & decoder. For feature reduction, compressing the input data at the encoding layer can reduce the number of nodes in the output layer. Finding the initial, unencoded data is what the decoding layer is all about. There is a perfect correspondence between the amount of input nodes & number of output nodes. Reconstruction errors can be discovered by direct input-to- output comparisons, & best AE models are developed through layer-by-layer training aimed at minimizing output loss.The complexity of detecting intrusions in networks is increasing as the nature of these attacks changes. Uneven network traffic makes it hard for IDS in particular to foresee the spread of malicious assaults. Thus, cyber security is a major issue. This study introduces a novel DSSTE method that recovers the classification model's ability to simplify to networks with uneven representations of nodes. To reduce network traffic disparity, it is possible to increase the number of alternative samples that need to be learned in order to improve classification accuracy by reducing network congestion. We employed six time-honored ML & DL classification strategies, combining them with various sample approaches. When there is asymmetry in network traffic, our method efficiently identifies which samples need to be enhanced to better attack recognition.

Using the DSSTE algorithm to draw from the unbalanced training set, we observed that DL performed better than ML. Since the data features in publicly available datasets have already been preprocessed, deep learning is given the capacity to learn the preprocessed features instead of relying on automatic feature extraction. Our objective is to use the DL model directly on the raw network traffic data for feature extraction & model training in order to take use of DL's benefits in this context, reduce the impact of imbalanced data, or recover the quality of the classification.

5.2 FUTURE SCOPE

The main focus at this point is on the potential of ML& DL to influence the future of IDS research.

The NIDS framework is essential in the fight against network intrusions, which is a major concern. New studies show that it is very prone to false alarms and cannot reliably identify zero-day attacks. This is why it's possible to boost IDS performance by switching to a more current, systematic, or evenly distributed dataset. Very little effort has been put forward by researchers to create an effective and all-encompassing network-wide NIDS architecture, especially for contemporary networks such as the IoT. A strong NIDS architecture that can provide impenetrable security may be the result of future research in this field. If an IDS framework wants to keep its model up- to-date, it must continually update the dataset's attack definitions. With fewer false positives and improved zero-day threat detection, the IDS model is ready to go. Training an IDS model can be arduous & time-consuming. Maintaining a steady stream of dataset updates & training is essential for AI-based IDS systems.

Excellent results in identifying malicious assaults can be obtained with DL-based IDSs because of their deep feature learning capabilities, which have recently gained a lot of attention from researchers. Developing models with DL techniques requires substantial investments in time, storage, and processing power. Installing IDS in real time is made more difficult by these intricate layouts. Fast and efficient analysis of massive datasets using high-performance GPUs is one approach to addressing these issues. However, such GPUs can be quite pricey. Therefore, finding a happy medium between efficiency and cost is essential. By utilising cloud-based GPU platforms or services, model training can be accomplished at a reduced cost. Furthermore, smart and efficient feature engineering has the potential to decrease the intricacy of DL algorithms. When all features are utilised, the detection accuracy is approximately the same as when only the most significant ones are utilised. Because of this, both the model's complexity & amount of processing power required in a real-time setting will be reduced. According to new analysis, DL-based algorithms should be

used when designing IDS. Prior to proposing a single DL approach for an IDS solution for an IoT network, additional investigate is required. Research on DL with an eye towards IDS is still in its early stages. Data classification & feature extraction might benefit from a combination of ML& DL. For that reason, the suggested framework will include the simplified model.

Cyber-Physical Systems that Do Not Invade Their Targets There has been a lot of recent focus on cyber-physical systems, which include SCADA networks and networks that can be used with UAVs. The use of SCADA networks is widespread in many industries, including smart grids and manufacturing. It's true that hackers can more easily breach a network as it grows increasingly integrated with information and communication technology (ICT). NIDS are vital because they can analyse network data and detect intrusions. SCADA network cyberattack detection is another application of ML & DL. We need additional research to find & develop actual DL-based NIDS for SCADA networks. Networks enabled by UAVs can do more than only safeguard critical infrastructure; they can also monitor roadways and examine assets. Intruders can listen in on or attack users of such networks because of the very nature of wireless communication. A high-tech NIDS is necessary for UAV-supporting networks to identify intrusions. More information regarding the potential applications of AI in the NIDS for networks enabled by UAVs would be fascinating.

REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A Survey," 2010, Computer Networks, vol. 54, no. 15, pp. 2787–2805.

[2] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, "Internet of things (IoT): A vision, architectural elements, and future directions," Elsevier Future Generation Computer System, Vol. 29, No. 7, pp. 1645–1660, 2013.

[3] I. Mashal, O. Alsaryrah, T.-Y. Chung, C.-Z. Yang, W.-H. Kuo, and D. P. Agrawal, "Choices for interaction with things on Internet and underlying issues," 2015, Ad Hoc Networks, vol. 28, pp. 68–90O. Said and M. Masud, "Towards internet of things: survey and future vision," 2013, International Journal of Computer Networks, vol. 5, no. 1, pp. 1–17.

[4] J. S. Kumar and D. R. Patel, "A survey on internet of things: Security and privacy issues," 2014, International Journal of Computer Applications, vol. 90, no. 11.

[5] R. M. Cardoso, N. Mastelari, and M. F. Bassora, "Internet of Things Architecture in the Context of Intelligent Transportation System – A Case Study Towards a Webbased Application Deployment," in 22nd International Congress of Mechanical Engineering (COBEM 2013), 2013, pp. 7751–7760.

[6] T. Abels, R. Khanna, K. Midkiff, "Future Proof IoT: Composable Semantics, Security, QoS and Reliability", 2017, in Proc. of IEEE Topical Conference on Wireless Sensor & Sensor Networks (WiSNet), pp. 1-4.

[7] Chandni, Rakesh Kumar, "Trust Based Technique for the Mitigation of Version Number Attack in Internet of Things", 2019, International Journal of Recent Technology and Engineering (IJRTE), Volume8 Issue3.

[8] M.V.R Jyothisree, S. Sreekanth, "Attacks in RPL and Detection Technique used for Internet of Things", 2019, International Journal of Recent Technology and Engineering (IJRTE), Volume8, Issue1.

[9] Abhishek Verma and Virender Ranga, “Analysis of Routing Attacks on RPL based 6LoWPAN Networks”, 2018, International Journal of Grid and Soft Computing, Volume 11, Issue 8, pp. 43-56.

[10] Ahmet Aris, Sema F. Oktug, S. Berna Ors Yalcin, “RPL version number attacks: In-depth study”, NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium.

[11] Ahmet Arış, Sema F. Oktuğ, “Analysis of the RPL Version Number Attack with Multiple Attackers”, 2020, International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA).

[12] Ş. Okul, M. Ali Aydın, “Security Attacks on IoT”, 2017, International Conference on Computer Science and Engineering (UBMK).

[13] Ruchi Vishwakarma, Ankit Kumar Jain, “A Honeypot with Machine Learning based Detection Framework for defending IoT based Botnet DDoS Attacks”, 2019, 3rd International Conference on Trends in Electronics and Informatics (ICOEI).

[14] Vijender Busi Reddy, Atul Negi, S Venkataraman, V Raghu Venkataraman, “A Similarity based Trust Model to Mitigate Badmouthing Attacks in Internet of Things (IoT)”, 2019, IEEE 5th World Forum on Internet of Things (WF-IoT).

[15] Xupeng Luo, Qiao Yan, Mingde Wang, Wenyao Huang, “Using MTD and SDN- based Honeypots to Defend DDoS Attacks in IoT”, 2019, Computing, Communications and IoT Applications (ComComAp).

[16] MeriemBettayeb, Omnia Abu Waraga, Manar Abu Talib, Qassim Nasir, Omar Einea, “IoT Testbed Security: Smart Socket and Smart Thermostat”, 2019 IEEE Conference on Application, Information and Network Security (AINS).

[17] Jalindar B. Karande, Sarang A. Joshi, “Comprehensive Assessment of Security Attack Detection Algorithms in Internet of Things”, 2018, Fourth

International Conference on Computing Communication Control and Automation (ICCUBEA).

[18] Lulu Liang, Kai Zheng, Qiankun Sheng, Xin Huang, "A Denial of Service Attack Method for an IoT System", 2016, 8th International Conference on Information Technology in Medicine and Education (ITME).

[19] Arjun Shakdher, Suyash Agrawal, Baijian Yang, "Security Vulnerabilities in Consumer IoT Applications", 2019, IEEE 5th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance.

[20] Salim Chehida, AbdelhakimBaouya, Marius Bozga, Saddek Bensalem, "Exploration of Impactful Countermeasures on IoT Attacks", 2020, 9th Mediterranean Conference on Embedded Computing (MECO).

[21] Ahmed Yar Khan, Rabia Latif, Seemab Latif, Shahzaib Tahir, Gohar Batool, Tanzila Saba, "Malicious Insider Attack Detection in IoTs Using Data Analytics", 2020, IEEE Access, Volume: 8.

[22] ZieEyaEkolle, KuramitsuKimio, Kohno Ryuji, "Intelligent Security Monitoring in Time Series of DDoS attack on IoT Networks using Grammar base Filtering and Clustering", 2018, International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS).

[23] Ali Hameed, AlauddinAlomary, "Security Issues in IoT: A Survey", 2019, International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT).

[24] MukrimahNawir, Amiza Amir, NaimahYaakob, Ong Bi Lynn, "Internet of Things (IoT): Taxonomy of security attacks", 2016, 3rd International Conference on Electronic Design (ICED).

[25] E. Abinaya, K Aishwarva, Cecil Prabhaker, M Lordwin, G Kamatchi ; I Malarvizhi, "A Performance Aware Security Framework to Avoid Software Attacks on Internet of Things (IoT) Based Patient Monitoring System", 2018,

International Conference on Current Trends towards Converging Technologies (ICCTCT).

[26] Kashif Naseer Qureshi, Shahid Saeed Rana, Awais Ahmed, Gwanggil Jeon, "A novel and secure attacks detection framework for smart cities industrial internet of things", 2020, Sustainable Cities and Society.

[27] Ning Zhang, Renyong Wu, Shenglan Yuan, Chao Yuan, Dajiang Chen, "RAV: Relay Aided Vectorized Secure Transmission in Physical Layer Security for Internet of Things Under Active Attacks", 2019, IEEE Internet of Things Journal, Volume: 6, Issue: 5.

[28] Yair Meidan, Michael Bohadana, Yael Mathov, Yisroel Mirsky, Asaf Shabtai, Dominik Breitenbacher, Yuval Elovici, "N-BaIoT—Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders", 2018, IEEE Pervasive Computing, Volume: 17, Issue: 3.

[29] Jyoti Deogirikar, AmarsinhVidhate, "Security attacks in IoT: A survey", 2017, International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud (I-SMAC).

[30] Seungyong Yoon, Jeongnyeo Kim, "Remote security management server for IoT devices", 2017, International Conference on Information and Communication Technology Convergence (ICTC).

[31] Max Ingham,JimsMarchang, Deepayan Bhowmik, "IoT security vulnerabilities and predictive signal jamming attack analysis in LoRaWAN", 2020, IET Information Security.

[32] A. Zanella N. Bui A. Castellani L. Vangelista and M. Zorzi "Internet of things for smart cities" IEEE Internet Things J. vol. 1, no. 1, pp. 22-32 ,2014.

[33] B. B. Gupta Computer and cyber security: principles algorithm applications and perspectives CRC, Press 2018.

[34] J. A. Jerkins "Motivating a market or regulatory solution to IoT insecurity with the Mirai botnet code" 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC) pp. 1-5, 2017.

[35] Mengmei Ye Nan Jiang Hao Yang and Qiben Yan “Security analysis of Internet of Things: A case study of august smart lock” 2017. IEEE Conference on Computer Communications Workshops (INFO-COM WKSHPS), pp. 499-504 ,2017.

[36] Nathaniel Gyory and M Chuah “IoTOne: Integrated platform for heterogeneous IoT devices” 2017, International Conference on Computing Networking and Communications (ICNC) pp.783-787 , 2017.

[37] Anton O Prokifiev, Yulia S Smirnova and Vasiliy A Surov "A method to detect Internet of Things botnets" 2018, IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), pp. 105-108,2018.

[38] Vinay Sachidananda, Shachar Siboni ,Asaf Shabtai ,Jinghui Toh ,Suhas Bhairav and Yuval Elovici, "Let the cat out of the bag: A holistic approach towards security analysis of the internet” Proceedings of the 3rd ACM International Workshop on IoT Privacy Trust and Security pp. 3-10, 2017.

[39] Sergei Chistiakov "Secure storage and transfer of data in a smart lock system".

[40] Mahmoud Ammar, Giovanni Russello and Bruno Crispo "Internet of Things: A survey on the security of IoT frameworks" in Journal of Information Security and Applications Elsevier, 2017.

[41] Thomas Willingham ,Cody Henderson, Blair Kiel, Md Shariful Haque and Travis Atkison "Testing vulnerabilities in bluetooth low energy" Proceedings of the ACMSE 2018, Conference 38, pp. 8-27 2018, pp. 6 2018.

[42] Z. Ling J. Luo Y. Xu C. Gao K. Wu and X. Fu "Security Vulnerabilities of Internet of Things: A Case Study of the Smart Plug System" IEEE Internet Things J. 2017.

[43] Y. Seralathan et al. "IoT security vulnerability: A case study of a Web camera" Advanced Communication Technology (ICACT), 2018, 20th International Conference on pp. 172-177 GLOBECOM 2017, IEEE Global Communications Conference pp. 1- 7, 2017-2018.

[44] H. Xu F. Xu and B. Chen "Internet Protocol Cameras with No Password Protection: An Empirical Investigation" International Conference on Passive and Active Network Measurement pp. 47-59, 2018.

[45] L. Huraj M. Simon and T. Horák "IoT Measuring of UDP-Based Distributed Reflective DoS Attack" 2018 IEEE 16th International Symposium on Intelligent Systems and Informatics SISY) pp. 209-214 ,2018.

[46] Siboni A. Shabtai and Y. Elovici "Leaking data from enterprise networks using a compromised smartwatch device" Proceedings of the 33rd Annual ACM Symposium on Applied Computing pp. 741-750, 2018.

[47] J. Classen D. Wegemer P. Patras T. Spink and M. Hollick "Anatomy of a Vulnerable Fitness Tracking System: Dissecting the Fitbit Cloud App and Firmware" Proc. ACM Interactive Mobile Wearable Ubiquitous Technol. vol. 2 no. 1 pp. 5 2018.

[48] Qiu, Tie, et al. "ERGID: An efficient routing protocol for emergency response Internet of Things." Journal of Network and Computer Applications 72 (2016): 104- 112.

[49] Debroy, Saptarshi, et al. "SpEED-IoT: Spectrum aware energy efficient routing for device-to-device IoT communication." Future Generation Computer Systems (2018).

[50] Pan, Meng-Shiuan, and Shu-Wei Yang. "A lightweight and distributed geographic multicast routing protocol for IoT applications." Computer Networks 112 (2017): 95- 107.

[51] Pan, Meng-Shiuan, and Shu-Wei Yang. "A lightweight and distributed geographic multicast routing protocol for IoT applications." Computer Networks 112 (2017): 95- 107.

[52] Ishino, Masanori, Yuki Koizumi, and Toru Hasegawa. "A study on routing-based mobility management architecture for IoT devices." 2014 IEEE 22nd International Conference on Network Protocols (ICNP). IEEE, 2014.

[53] Otermat, Derek T., Carlos E. Otero, and Ivica Kostanic. "Analysis of the FM radio spectrum for Internet of Things opportunistic access via cognitive radio." Internet of Things (WF-IoT), 2015 IEEE 2nd World Forum on. IEEE, 2015.

[54] Huang, Jun, et al. "Multicast routing for multimedia communications in the Internet of Things." IEEE Internet of Things Journal 4.1 (2017): 215-224.

[55] Hasan, Mohammed Zaki, and Fadi Al-Turjman. "Optimizing multipath routing with guaranteed fault tolerance in Internet of Things." IEEE Sensors Journal 17.19 (2017): 6463-6473.

[56] Feng, Zhiyong, et al. "Priority-based dynamic spectrum management in a smart grid network environment." IEEE Journal on Selected Areas in Communications 33.5 (2015): 933-945.

[57] Khan, Athar Ali, Mubashir Husain Rehmani, and Martin Reisslein. "Requirements, design challenges, and review of routing and MAC protocols for CR- based smart grid systems." IEEE Communications Magazine 55.5 (2017): 206-215.

[58] Bello, Oladayo, and SheraliZeadally. "Intelligent device-to-device communication in the internet of things." IEEE Systems Journal 10, no. 3 (2016): 1172-1182.

[59] Wang, Xiaofei, Xiuhua Li, and Victor CM Leung. "Artificial intelligence-based techniques for emerging heterogeneous network: State of the arts, opportunities, and challenges." IEEE Access 3 (2015): 1379-1391.

[60] Sharma, Vishal, Fei Song, Ilsun You, and Mohammed Atiquzzaman. "Energy efficient device discovery for reliable communication in 5G-based IoT and BSNs using unmanned aerial vehicles." Journal of Network and Computer Applications 97 (2017): 79-95.

[61] Meng, Yue, Chunxiao Jiang, Hsiao-Hwa Chen, and Yong Ren. "Cooperative device-to-device communications: Social networking perspectives." IEEE Network 31, no. 3 (2017): 38-44.

[62] Li, Xiaomin, Di Li, Jiafu Wan, Chengliang Liu, and Muhammad Imran. "Adaptive transmission optimization in SDN-based industrial Internet of Things with edge computing." IEEE Internet of Things Journal 5, no. 3 (2018): 1351-1360.

[63] Dayal, N., Maity, P., Srivastava, S., & Khondoker, R. (2016). Research trends in security and DDoS in SDN. Security and Communication Networks, 9(18), 6386- 6411.

[64] R. H. Jhaveri, N. M. Patel, Y. Zhong, & A. K. Sangaiah, ―Sensitivity analysis of an attack-pattern discovery based trusted routing scheme for mobile ad-hoc networks in industrial IoT,‖ IEEE Access, vol. 6, pp. 20085-20103, 2018.

[65] Ma, H., Liu, L., Zhou, A., & Zhao, D. (2016). On networking of internet of things: Explorations and challenges. IEEE Internet of Things Journal, 3(4), 441-452.

[66] N. Arya, U. Singh, and S. Singh,‖ Detecting and avoiding of wormhole attack and collaborative black hole attack on MANET using trusted AODV routing algorithm. In 2015 International Conference on Computer, Communication, and Control (IC4) IEEE, pp. 1-5, 2015.

[67] P. Gupta, P. Goel, P. Varshney, and N. Tyagi, ―Reliability Factor-Based AODV Protocol: Prevention of Black Hole Attack in MANET,‖ In Smart Innovations in Communication and Computational Sciences, Springer, Singapore, pp. 271-279, 2019.

[68] T. A. Kolade, "A Scheme for detecting and mitigating cooperative black hole attack in AODV-based MANET routing protocol." Ph.D. dissertation, 2018.

[69] A. Adnan, A. B. Kamalrulniza, M. I. Channa, and A.W. Khan. "A secure routing protocol with trust and energy awareness for wireless sensor

network," Mobile Networks and Applications, Vol. 21, No. 2. pp 272-285, 2016.

[70] Jain, A. K., Tokekar V. &Shrivastava S., ―Security Enhancement in MANETs Using Fuzzy-Based Trust Computation Against Black Hole Attacks, in Information and Communication Technology, pp. 39-47, Springer, Singapore, 2018.

[71] Chhabra, A., Vashishth, V., & Sharma, D. K., ―A fuzzy logic and game theory based adaptive approach for securing opportunistic networks against black hole Attacks,‖ International Journal of Communication Systems, vol. 31, no. 4, pp. 3487- 3510, 2018.

[72] Abdel-Azim, M., Salah, H. E. D., &Eissa, ―IDS Against Black-Hole Attack for MANET,‖ IJ Network Security, vol. 20 no. 3, pp. 585-592, 2018.

[73] Gurung, S., &Chauhan, S, ―A novel approach for mitigating gray hole attack in MANET. Wireless Networks, vol. 24, No.2, pp.565-579, 2019.

[74] Trivedi, M. C., &Malhotra, S, ―Identification and Prevention of Joint Gray Hole and Black Hole Attacks,‖ International Journal of Ambient Computing and Intelligence (IJACI), vol. 10, no. 2, pp.80-90, 2019.

[75] Ali Zardari, Z., He, J., Zhu, N., Mohammadi, K. H., Pathan, M. S., Hussain,

M. I., &Memon, . Q, ―A Dual Attack Detection Technique to Identify Black and Gray Hole Attack Using an Intrusion Detection System and a Connected Dominating Set in MANETs,‖ Future Internet, vol. 11, no. 3, pp.61-78, 2019.

[76] Saudi, N. A. M., Arshad, M. A., Buja, A. G., Fadzil, A. F. A., & Saidi, R. M, ―Mobile Ad-Hoc Network (MANET) Routing Protocols: A Performance Assessment,‖ In Proceedings of the Third International Conference on Computing, Mathematics and Statistics (iCMS2017) (pp. 53-59). Springer, Singapore.

[77] Khanpara, P., &Trivedi, B, ―Security in mobile ad hoc networks. In Proceedings of International Conference on Communication and Networks (pp. 501- 511). Springer, Singapore.

[78] Kondaiah, R., &Sathyanarayana, B. (2018). Trust-based Genetic Neuro-Fuzzy System for Intrusion Detection and Self Adaptive Firefly integrated Particle Swarm Optimization Algorithm for Secure Routing in MANET. International Journal of Applied Engineering Research, 13(8), 5722-5735.

[79] Shona, D., & Kumar, M. S. (2018, July). Efficient IDs for MANET Using Hybrid Firefly with a Genetic Algorithm. In 2018 International Conference on Inventive Research in Computing Applications (ICIRCA) (pp. 191-194). IEEE.

[80] Zakaria, A. H., Saman, M. Y. M., Noor, A. S. M. & Hassan, H. (2015). Finding the shortest routing solution in mobile ad hoc networks using the firefly algorithm and queuing network analysis. Jurnal Teknologi, 77(18).

[81] Patel, N. J., &Jhaveri, R. H. (2015). Detecting Packet Dropping Misbehaving Nodes using Support Vector Machine (SVM) in MANET. International Journal of Computer Applications, 122(4).

[82] Mukesh, K. G., Khanna, H. P., &Velvizhi, R. V. (2015). An anomaly-based Intrusion Detection System for mobile ad-hoc networks using a genetic algorithmbased support vector machine. Advances in Natural and Applied Sciences, 9(12), 40-45.

[83] Brindha, V., Karthikeyan, T., &Manimegalai, P. (2019). Fuzzy enhanced secure multicast routing for improving authentication in MANET. Cluster Computing, 22(4), 9615-9623.

[84] Jamaesha, S. S., &Bhavani, S. (2019). A secure and efficient cluster-based location-aware routing protocol in MANET. Cluster Computing, 22(2), 4179-4186.

[85] Balamurugan, K., Chitra, K., &Jawahar, A. (2018). Enhanced hierarchical clusterbased routing protocol with optical sphere in FSO MANET. In Optical and Microwave Technologies (pp. 1-8). Springer, Singapore.

[86] Sethuraman, P., &Kannan, N. (2017). Refined trust energy-ad hoc on demand distance vector (ReTE-AODV) routing algorithm for secured routing in MANET. Wireless Networks, 23(7), 2227-2237.

[87] Ragsdale, Daniel J., et al. "Adaptation techniques for intrusion detection and intrusion response systems." Systems, Man, and Cybernetics, 2000 IEEE International Conference on. Vol. 4. IEEE, 2000.

[88] Wu, Zheng, et al. "Automated intrusion response decision based on the analytic hierarchy process." Knowledge Acquisition and Modeling Workshop, 2008. KAM Workshop 2008. IEEE International Symposium on. IEEE, 2008.

[89] Jensen, Finn V. An introduction to Bayesian networks. Vol. 210. London: UCL press, 1996.

[90] Zhang, Yongguang, Wenke Lee, and Yi-An Huang. "Intrusion detection techniques for mobile wireless networks." Wireless Networks 9.5 (2003): 545-556.

[91] Patcha, Animesh, and Jung-Min Park. "An overview of anomaly detection techniques: Existing solutions and latest technological trends." Computer networks 51.12 (2007): 3448-3470.

[92] Singh, Amrit Pal, and Manik Deep Singh. "Analysis of host-based and network- based intrusion detection system." International Journal of Computer Network and Information Security 6.8 (2014): 41-47.

[93] Juszczyszyn, Krzysztof, et al. "Agent-based approach for distributed intrusion detection system design." International Conference on Computational Science. Springer, Berlin, Heidelberg, 2006.

[94] CAI, Zeng-yu, et al. "Development of intelligent intrusion detection based on biosimulation [J]." Journal of Zhengzhou University of Light Industry (Natural Science) 2 (2010): 018.

[95] Xiaodong, W., et al. "Development of a snort-based security network management and real-time intrusion detection system." Journal-Beijing Normal University Natural Science Edition 40.1 (2004): 40-43.

[96] Datasets, Resilient Distributed. "A Fault-Tolerant Abstraction for In-Memory Cluster Computing." Matei Zaharia, Mosharaf Chowdhury, Tathagata Das, Ankur Dave, Justin Ma, Murphy McCauley, Michael J. Franklin, Scott Shenker, Ion Stoica. NSDI (2012).

[97] https://kdd.ics.uci.edu/databases/kddcup99/task.html, kddcup 2/26/2014.

[98] Shiravi, Ali, et al. "Toward developing a systematic approach to generate benchmark datasets for intrusion detection." computers & security 31.3 (2012): 357-374.

[99] Sharafaldin, Iman, Arash Habibi Lashkari, and Ali A. Ghorbani. "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization." ICISSP. 2018.

[100] Zikopoulos, Paul, and Chris Eaton. Understanding big data: Analytics for enterprise class hadoop and streaming data. McGraw-Hill Osborne Media, 2011.

[101] Sung, Andrew H., and Srinivas Mukkamala. "Identifying important features for intrusion detection using support vector machines and neural networks." Applications and the Internet, 2003. Proceedings. 2003 Symposium on. IEEE, 2003.

[102] Georges, Jim, and Anne H. Milley. "Kdd'99 competition: Knowledge discovery contest." ACM SIGKDD Explorations Newsletter 1.2 (2000): 79-84.

[103] John, George H., Ron Kohavi, and Karl Pfleger. "Irrelevant features and the subset selection problem." Machine Learning Proceedings 1994. 1994. 121-129.

[104] Dong, Shi, DingDing Zhou, and Wei Ding. "The study of network traffic identification based on machine learning algorithm." Computational Intelligence and Communication Networks (CICN), 2012 Fourth International Conference on. IEEE, 2012.

[105] Kulariya, Manish, et al. "Performance analysis of network intrusion detection schemes using Apache Spark." Communication and Signal Processing (ICCSP), 2016 International Conference on. IEEE, 2016.

[106] Bandre, Sanraj Rajendra, and Jyoti N. Nandimath. "Design consideration of Network Intrusion detection system using Hadoop and GPGPU." Pervasive Computing (ICPC), 2015 International Conference on. IEEE, 2015.

[107] Xiang, Junlong, et al. "Using extreme learning machine for intrusion detection in a big data environment." Proceedings of the 2014 Workshop on Artificial Intelligent and Security Workshop. ACM, 2014.

[108] Hu, Weiming, et al. "Online adaboost-based parameterized methods for dynamic distributed network intrusion detection." IEEE Transactions on Cybernetics 44.1 (2014): 66-82.

[109] Jabez, J., and B. Muthukumar. "Intrusion Detection System (IDS): Anomaly detection using outlier detection approach." Procedia Computer Science 48 (2015): 338-346.

[110] Chen, Rung-Ching, et al. "Using rough set and support vector machine for network intrusion detection system." Intelligent Information and Database Systems, 2009. ACIIDS 2009. First Asian Conference on. IEEE, 2009.

[111] Hadi, Theyazn Hassn, and Manish R. Joshi. "Handling ambiguous packets in intrusion detection." Signal Processing, Communication, and Networking (ICSCN), 2015 3rd International Conference on. IEEE, 2015.

[112] Takkellapati, Venkata Suneetha, and G. V. S. N. R. V. Prasad. "Network intrusion detection system based on feature selection and triangle area support vector machine." International Journal of Engineering Trends and Technology3.4 (2012).

[113] Upasani, Nilam, and Hari Om. "Evolving fuzzy min-max neural network for outlier detection." Procedia computer science 45 (2015): 753-761.

[114] Shanmugam, Bharanidharan, and Norbik Bashah Idris. "Improved intrusion detection system using fuzzy logic for detecting anomaly and misuse type of attacks." Soft Computing and Pattern Recognition, 2009. SOCPAR'09. International Conference of. IEEE, 2009.

[115] Shanmugavadivu, R., and N. Nagarajan. "Network intrusion detection system using fuzzy logic." Indian Journal of Computer Science and Engineering (IJCSE) 2.1 (2011): 101-111.

[116] Azad, Chandrashekhar, and Vijay Kumar Jha. "Fuzzy min–max neural network and particle swarm optimization based intrusion detection system." Microsystem Technologies 23.4 (2017): 907-918.

[117] Singh, Shubhangi, and Rajendra Singh Kushwah. "A Study on Intrusion Detection in Wireless Networks by Using Genetic Algorithm Applications." Computational Intelligence and Communication Networks (CICN), 2014 International Conference on. IEEE, 2014.

[118] Benaicha, Salah Eddine, et al. "Intrusion detection system using genetic algorithm." Science and Information Conference (SAI), 2014. IEEE, 2014.

[119] Devarakonda, Nagaraju, et al. "Intrusion detection system using bayesian network and hidden markov model." Procedia Technology 4 (2012): 506-514.

[120] Altwaijry, Hesham, and Saeed Algarny. "Bayesian based intrusion detection system." Journal of King Saud University-Computer and Information Sciences 24.1 (2012): 1-6.

[121] Almansob, Saqr Mohammed, and Santosh Shivajirao Lomte. "Addressing challenges for intrusion detection system using naive Bayes and PCA algorithm." Convergence in Technology (I2CT), 2017 2nd International Conference for. IEEE, 2017.

[122] https://kdd.ics.uci.edu/databases/kddcup99/task.html, kddcup 2/26/2014.

[123] Jonnalagadda, Sai Jyothsna. Global Entropy Based Greedy Algorithm for discretization. The University of Texas Rio Grande Valley, 2016.

[124] Ho, K. M., and P. D. Scott. "Zeta: a global method for discretization of cotitinuous variables." Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining. 1997.

[125] Wong, Andrew KC, and David KY Chiu. "Synthesizing statistical knowledge from incomplete mixed-mode data." IEEE Transactions on Pattern Analysis and Machine Intelligence 6 (1987).

[126] Quinlan, J. Ross. "Learning efficient classification procedures and their application to chess end games." Machine learning. Springer, Berlin, Heidelberg, 1983. 463-482.

[127] Clark, Peter, and Tim Niblett. "The CN2 induction algorithm." Machine learning 3.4 (1989): 261-283.

[128] Pham, D. T., and A. A. Afify. "Online discretization of continuous-valued attributes in rule induction." Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 219.8 (2005): 829-842.

[129] Kotsiantis, Sotiris, and Dimitris Kanellopoulos. "Discretization techniques: A recent survey." GESTS International Transactions on Computer Science and Engineering 32.1 (2006): 47-58.

[130] Aggarwal, Megha. "Performance analysis of different feature selection methods in intrusion detection." International Journal of Scientific & Technology Research 2.6 (2013): 225-231.

[131] Gennari, John H., Pat Langley, and Doug Fisher. "Models of incremental concept formation." Artificial intelligence 40.1-3 (1989): 11-61.

[132] Kohavi, Ron, and George H. John. "Wrappers for feature subset selection." Artificial intelligence 97.1-2 (1997): 273-324.

[133] Hall, Mark Andrew. "Correlation-based feature selection for machine learning." (1999).

[134] Kumar, Koushal, and Jaspreet Singh Batth. "Network intrusion detection with feature selection techniques using machine-learning algorithms." International Journal of Computer Applications 150.12 (2016).

[135] Yu, Lei, and Huan Liu. "Feature selection for high-dimensional data: A fast correlation-based filter solution." Proceedings of the 20th international conference on machine learning (ICML-03). 2003.

[136] Tian, Shengfeng, Shaomin Mu, and Chuanhuan Yin. "Sequence-similarity kernels for SVMs to detect anomalies in system calls." Neurocomputing 70.4-6 (2007): 859-866.

[137] Yao, JingTao, Songlun Zhao, and Lisa Fan. "An enhanced support vector machine model for intrusion detection." International Conference on Rough Sets and Knowledge Technology. Springer, Berlin, Heidelberg, 2006.

[138] Chen, Rongchang, et al. "Building an intrusion detection system based on support vector machine and genetic algorithm." International Symposium on Neural Networks. Springer, Berlin, Heidelberg, 2005.

[139] Tsang, Ivor W., James T. Kwok, and Pak-Ming Cheung. "Core vector machines: Fast SVM training on very large data sets." Journal of Machine Learning Research 6.Apr (2005): 363-392.

[140] Wafa, S. "Development of genetic-based machine learning for network intrusion detection." World Academy of Science, Engineering and Technology 55 (2009): 20-24.

[141] McHugh, John. "Testing intrusion detection systems: a critique of the 1998 and 1999 darpa intrusion detection system evaluations as performed by lincoln laboratory." ACM Transactions on Information and System Security (TISSEC)3.4 (2000): 262-294.

[142] Mulak, Punam, and Nitin Talhar. "Analysis of Distance Measures Using K-Nearest Neighbor Algorithm on KDD Dataset." International Journal of Science and Research 4.7 (2015): 2101-2104.

[143] Hernández-Pereira, Elena, et al. "Conversion methods for symbolic features: A comparison pplied to an intrusion detection problem." Expert Systems with Applications 36.7 (2009): 10612-10617.

[144] Modi, Chirag N., et al. "Bayesian Classifier and Snort based network intrusion detection system in cloud computing." 2012 Third International Conference on Computing, Communication and Networking Technologies (ICCCNT 2012). IEEE, 2012.

[145] Zuech, Richard, Taghi M. Khoshgoftaar, and Randall Wald. "Intrusion detection and big heterogeneous data: a survey." Journal of Big Data 2.1 (2015): 3.

[146] Spark, A. Spark Homepage http://spark.apache.org.

[147] Tan, Zhiyuan, et al. "Enhancing big data security with collaborative intrusion detection." IEEE cloud computing 3 (2014): 27-33.

[148] Oral, H. Sarp, et al. Real-Time System Log Monitoring/Analytics Framework. Oak Ridge National Laboratory (ORNL); Center for Computational Sciences, 2011.

[149] Solaimani, Mohiuddin, Latifur Khan, and Bhavani Thuraisingham. "Real-time anomaly detection over VMware performance data using storm." 2014 IEEE International Conference on Information Reuse and Integration (IRI). IEEE, 2014.

[150] Prathibha, P. G., and E. D. Dileesh. "Design of a hybrid intrusion detection system using snort and hadoop." International Journal of Computer Applications 73.10 (2013).

[151] Jeong, Hae-Duck J., et al. "Anomaly teletraffic intrusion detection systems on hadoop-based platforms: A survey of some problems and solutions." Network-Based Information Systems (NBiS), 2012 15th International Conference on. IEEE, 2012.

[152] Solaimani, Mohiuddin, et al. "Spark-based anomaly detection over multi-source VMware performance data in real-time." Computational Intelligence in Cyber Security (CICS), 2014 IEEE Symposium on. IEEE, 2014.

[153] Tangsatjatham, Pittayut, and Natawut Nupairoj. "Hybrid big data architecture for high-speed log anomaly detection." Computer Science and Software Engineering (JCSSE), 2016 13th International Joint Conference on. IEEE, 2016.

[154] Hasan, Md Al Mehedi, et al. "Support vector machine and random forest modeling for intrusion detection system (IDS)." Journal of Intelligent Learning Systems and Applications 6.01 (2014): 45.

[155] Ryza, Sandy, et al. Advanced analytics with spark: patterns for learning from data at scale. " O'Reilly Media, Inc.", 2017.

[156] Koliopoulos, Aris-Kyriakos, et al. "A parallel distributed weka framework for big data mining using spark." Big Data (BigData Congress), 2015 IEEE International Congress on. IEEE, 2015.

[157] Rathore, M. Mazhar, et al. "Hadoop based real-time intrusion detection for high-speed networks." Global Communications Conference (GLOBECOM), 2016 IEEE. IEEE, 2016.

[158] Alam, Farhad, and Sanjay Pachauri. "Detection using WEKA." Advances in Computational Sciences and Technology 10.6 (2017): 1731-1743.

[159] Aldubai, Ahmed Faud, Vikas T. Humbe, and Santosh S. Chowhan. "Analytical Study of Intruder Detection System in Big Data Environment." Soft Computing: Theories and Applications. Springer, Singapore, 2018. 405-416.

[160] Alheeti, Khattab M. Ali, Anna Gruebler, and Klaus D. McDonald-Maier. "An intrusion detection system against black hole attacks on the communication network of self-driving cars." Emerging Security Technologies (EST), 2015 Sixth International Conference on. IEEE, 2015.

[161] Jabez, J., and B. Muthukumar. "Intrusion Detection System (IDS): Anomaly detection using outlier detection approach." Procedia Computer Science 48 (2015): 338-346.

[162] Modi, Chirag N., et al. "Bayesian Classifier and Snort based network intrusion detection system in cloud computing." 2012 Third International Conference on Computing, Communication and Networking Technologies (ICCCNT 2012). IEEE, 2012.

[163] Ahmed Fuad Mohammed, Vikas T. Humbe, Santosh S. Chowhan, "Data Mining Based Traffic Classification Using Low-Level Features" International Journal of Computer Applications (0975 – 8887), 2017.

[164] Azad, Chandrashekar, and Vijay Kumar Jha. "Data mining based hybrid intrusion detection system." Indian Journal of Science and Technology 7.6 (2014): 781-789.

[165] Chen, Tieming, et al. "Efficient classification using parallel and scalable compressed model and its application on intrusion detection." Expert Systems with Applications 41.13 (2014): 5972-5983.

[166] Gu, Baoping, and Honyan Guo. "The intrusion detection system based on a novel association rule." Information Science, Electronics and Electrical Engineering (ISEEE), 2014 International Conference on. Vol. 2. IEEE, 2014.

[167] Quteishat, Anas, and Chee Peng Lim. "A modified fuzzy min–max neural network with rule extraction and its application to fault detection and classification." Applied Soft Computing 8.2 (2008): 985-995.

[168] Liao, Shu-Hsien, Pei-Hui Chu, and Pei-Yuan Hsiao. "Data mining techniques and applications–A decade review from 2000 to 2011." Expert systems with applications 39.12 (2012): 11303-11311.

[169] Barbará, Daniel, and Sushil Jajodia, eds. Applications of data mining in computer security. Vol. 6. Springer Science & Business Media, 2002.

[170] Zadeh, Lotfi Asker. "Fuzzy logic." Computer 21.4 (1988): 83-93.

[171] Simpson, Patrick K. "Fuzzy min-max neural networks. I. Classification." IEEE transactions on neural networks 3.5 (1992): 776-786.

[172] Anjay Krishnankutty Alonso, “eMath teacher for MAMBANI'S FUZZY INFERENCEMMETHOD”, Retrievedfrom,http://www.dma.fi.upm.es/recursos/aplicaciones/logica_borrosa/web/fuzzy_inferencia/funpert_en.htm, (2017, Oct, 19)

[173] Pfahringer, Bernhard. "Winning the KDD99 classification cup: bagged boosting." ACM SIGKDD Explorations Newsletter 1.2 (2000): 65-66.

[174] Levin, Itzhak. "KDD-99 classifier learning contest: LLSoft's results overview." SIGKDD explorations 1.2 (2000): 67-75.

[175] Kulkarni, Uday V., and T. R. Sontakke. "Fuzzy hypersphere neural network classifier." Fuzzy Systems, 2001. The 10th IEEE International Conference on. Vol. 3. IEEE, 2001.

[176] Kwan, Hon Keung, and Yaling Cai. "A fuzzy neural network and its application to pattern recognition." IEEE transactions on Fuzzy Systems 2.3 (1994): 185-193.

[177] Kulkarni, U. V., T. R. Sontakke, and G. D. Randale. "Fuzzy hyperline segment neural network for rotation invariant handwritten character recognition." Neural Networks, 2001. Proceedings. IJCNN'01. International Joint Conference on. Vol. 4. IEEE, 2001.

[178] Hou, Yong, and Xue Feng Zheng. "SVM based MLP neural network algorithm and application in intrusion detection." International Conference on Artificial Intelligence and Computational Intelligence. Springer, Berlin, Heidelberg, 2011.

[179] A hybrid supervised/unsupervised approach to network anomaly detection Gepostet am: 04. Oktober 2017 Julian Keppel und Sascha Schmalz.

[180] Zhang, Hong, and Xinghui Zhang. "Intrusion Detection Based on Improvement of Genetic Fuzzy C-Means Algorithm." Advances in Information Technology and Industry Applications. Springer, Berlin, Heidelberg, 2012. 339-346.

[181] Natesan, P., P. Balasubramanie, and G. Gowrison. "Performance comparison of AdaBoost based weak classifiers in network intrusion detection." Journal of Information Systems and Communication 3.1 (2012): 295.

[182] Witten, Ian H., et al. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, 2016.

[183] Tran, Tich Phuoc, et al. "Novel intrusion detection using probabilistic neural network and adaptive boosting." arXiv preprint arXiv:0911.0485 (2009).

[184] Wang, Gang, et al. "A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering." Expert systems with applications 37.9 (2010): 6225-6232.

[185] Yi, Yang, Jiansheng Wu, and Wei Xu. "Incremental SVM based on reserved set for network intrusion detection." Expert Systems with Applications 38.6 (2011): 7698-7707.

[186] Sindhu, Siva S. Sivatha, S. Geetha, and Arputharaj Kannan. "Decision tree based light weight intrusion detection using a wrapper approach." Expert Systems with applications 39.1 (2012): 129-141.

[187] Chandrashekhar, A. M., and K. Raghuveer. "Fortification of hybrid intrusion detection system using variants of neural networks and support vector machines." International Journal of Network Security & Its Applications 5.1 (2013): 71.

[188] Shrivas, Akhilesh Kumar, and Amit Kumar Dewangan. "An ensemble model for classification of attacks with feature selection based on KDD99 and NSL-

KDD data set." International Journal of Computer Applications 99.15 (2014): 8-13.

[189] Qiu, Chunmin, Jie Shan, and B. Shandong. "Research on intrusion detection algorithm based on BP neural network." International Journal of Security and its Applications 9.4 (2015): 247-258.

[190] Chandrashekhar, Azad, and Jha Vijay Kumar. "Fuzzy Min-Max Neural Network-Based Intrusion Detection System." Proceedings of the International Conference on Nano-electronics, Circuits & Communication Systems. Springer, Singapore, 2017.

PUBLICATION DETAILS

1. Zakiya, M., & Harjit, S. (26 July 2023). Deep Neural Network Solution for Detecting Intrusion in Network. In *International Journal on Recent and Innovation Trends in Computing and Communication*. ISSN: 2321- 8169, (Vol. 11 Issue. 8). **[Scopus Indexed]**

2. Zakiya, M., & Harjit, S. (28 October 2023). Network Intrusion Detection Using Autoencode Neural Network. In *International Journal on Recent and Innovation Trends in Computing and Communication*. ISSN: 2321-8169, (Vol. 11 Issue. 10). **[Scopus Indexed]**

3. Zakiya, M., & Harjit, S. (22 October 2023). Trust Based Mechanism for Isolation of Malicious Nodes in Internet of Things. In *International Journal on Recent and Innovation Trends in Computing and Communication*. ISSN: 2321-8169, (Vol. 11 Issue. 10). **[Scopus Indexed]**

4. Zakiya, M., & Harjit, S. (30 September 2022). Optimization Algorithms with Machine Learning to Improve Security of Internet of Things. In *International Journal on Recent and Innovation Trends in Computing and Communication*. ISSN: 2321-8169, (Vol. 11 Issue. 9). **[Scopus Indexed]**

5. Zakiya, M., (2-3 September 2023) presented a paper entitled: "The trust based model for intrusion detection in IOT" in an International conference on "Humanities, Computer, Management & Health" (NSHCMH-2023) organized by IAJESM.

6. Zakiya, M., (2-3 September 2023) presented a paper entitled: "Network Traffic Classification Using Machine Learning Models." in an International conference on "*Advances in Multidisplinary Research & Innovation" (ICAMRI-2023)* organized by IAJESM.