INTRUSION DETECTION USING DEEP LEARNING IN IOT-BASED SMART HEALTHCARE

By

ANA SHAHID

NATIONAL UNIVERSITY OF MODERN LANGUAGES

ISLAMABAD

AUGUST 2022

INTRUSION DETECTION USING DEEP LEARNING IN IOT-BASED SMART HEALTHCARE

By

ANA SHAHID

BSIT, University of Gujrat Sub campus, Rawalpindi, 2015

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE
In **Computer Science**

To
FACULTY OF ENGINEERING & COMPUTER SCIENCE

NATIONAL UNIVERSITY OF MODERN LANGUAGES ISLAMABAD

NATIONAL UNIUVERSITY OF MODERN LANGUAGES FACULTY OF ENGINEERIING & COMPUTER SCIENCE

THESIS AND DEFENSE APPROVAL FORM

The undersigned certify that they have read the following thesis, examined the defense, are satisfied with overall exam performance, and recommend the thesis to the Faculty of Engineering and Computer Sciences for acceptance.

Thesis Title: Intrusion Detection using Deep Learning in IoT Based Smart Healthcare

Submitted By: Ana Shahid **Registration #:** 36 MS/CS/S20

Master of Science in Computer Science (MSCS)
Title of the Degree

Computer Science
Name of Discipline

Dr. Sajjad Haider Name of Research Supervisor	Signature of Research Supervisor
Dr. Basit Shahzad Name of Dean (FE&CS)	Signature of Dean (FE&CS)
Prof. Dr. Muhammad Safeer Awan Name of Pro-Rector Academics	Signature of Pro-Rector Academics

August 30th 2022

AUTHOR'S DECLARATION

I Ana Shahid

Daughter of Muhammad Shahid

Registration # 36/MS/CS/S20

Discipline Computer Science

Candidate of **Master of Science in Computer Science (MSCS)** at the National University of Modern Languages do hereby declare that the thesis **Intrusion Detection Using Deep Learning in IoT-based Smart Healthcare** submitted by me in partial fulfillment of MSCS degree, is my original work, and has not been submitted or published earlier. I also solemnly declare that it shall not, in the future, be submitted by me for obtaining any other degree from this or any other university or institution. I also understand that if evidence of plagiarism is found in my thesis/dissertation at any stage, even after the award of a degree, the work may be cancelled and the degree revoked.

Signature of Candidate

Ana Shahid
Name of Candidate

30th August 2022
Date

ABSTRACT

Title: Intrusion Detection using Deep Learning in IoT based Smart Healthcare

The rapid increase and implementation of Internet of things (IoT) based technologies in healthcare have made a significant contribution to the global network. Despite bringing useful benefits all over the globe such as real-time monitoring of patients' information and diagnosing properly whenever needed, Internet of things (IoT) based systems appear to be an easy target for intruders. As the number of threats and attacks against IoT devices and services rapidly increases, the security of Internet of Things (IoT) in healthcare has become more challenging. In order to meet this challenge, hybrid learning based effective Intrusion Detection in IoT needs to be developed. In this study, we propose a novel hybrid model for intrusion detection in IoT based smart healthcare using RF, SVM, LSTM and gradient boosting. We proposes generalized model by handling the problems of overfitting and underfitting. We generates a new feature to make the proposed model more effective for detecting intrusion in IoT. We study the performance of proposed model in multi classification using MQTT-IOT-IDS 2020 dataset, a latest dataset with IoT network traces and compared the performance with different ML and DL algorithms. Experimental results show that our model performs better intrusion detection than other DL and ML algorithms.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

IoT	-	Internet of Things
AI		Artificial Intelligence
ML	-	Machine Learning
DL	-	Deep Learning
DoS		Denial of Service
DDoS		Distributed Denial of Service
U2R		User to Root
R2L		Resource to Locator
MiTM		Man in The Middle
IDS		Intrusion Detection System
ID		Intrusion Detection
IoMT		Internet of Medical Things
NoD		Network of Drones
RF	-	Random Forest
SVM		Support Vector Machine
LSTM		Long Short Term Memory
DT		Decision Tree
GNB		Gaussian Naïve Bayes
KNN		K-Nearest Neighbor
LR		Logistic Regression
GA		Genetic Algorithm
PCA		Principal Component Analysis
GB		Gradient Booster
XGB		Extreme Gradient Booster
CNN		Convolutional Neural Network
MLP		Multi-Layer Perceptron
GRU		Gated Recurrent Unit
RBM		Restricted Boltzmann Machine
DBN		Deep Belief Network
AE		Autoencoder
CAE		Convolutional Autoencoder
DAE		Denoising Autoencoder

SAE		Stacked Autoencoder
IGR		Information Gain Ratio
CFS		Correlation-based Feature Selection
SDN		Software Defined Network
ANN		Artificial Neural Network
FFNN		Feed Forward Neural Network
OC-SVM		One-Class Support Vector Machine
MQTT		Message Queue Telemetry Transfer

ACKNOWLEDGMENT

First and foremost, praises and thanks to Allah Almighty, for His showers of blessings throughout my thesis to complete this study successfully.

This endeavor would not have been possible without my research supervisor, Asst. Prof. Dr. Sajjad Haider, who gave me the opportunity to do research and provided worthy guidance throughout this study. It was a great privilege and honor to work and study under his guidance.

I shall also acknowledge the extended assistance from the Department of Computer Sciences administrations who supported me all through my research experience and simplified the challenges I faced. Lastly, I would be remiss in not mentioning my family, especially my parents, siblings, and friends.

DEDICATION

Every challenging task requires self-endeavor and the guidance of elders, especially those very close to our hearts.

My little effort I dedicate to my beloved

Parents

Whose love support and payers of days and night make me able to get such success and honor.

Along with all the hardworking and respected

Teacher

INTRODUCTION

1.1 Overview

The modern world is tremendously advancing toward the digitization of gadgets and systems by using the internet for running day-to-day operations [1]. However, for the development and deployment of the latest digital technologies such as Artificial intelligence (AI), 5g, Virtual Reality, etc., the internet alone cannot fulfill the requirements of these technologies. Nowadays, the Internet of Things (IoT) embeds with devices, software, sensors, actuators, storage, and computational capabilities to deal with the real-time environment and monitor the environment with precision. With the growing involvement of IoT, the traditional way of living is turning into a modern lifestyle. During the last decade, IoT has rapidly evolved in just about every technological area such as homes, healthcare [2][3], cities, transportation, grids [4], industries, etc. . Before the advent of IoT, medical services were just limited to telephone calls and visits. The IoT-based smart healthcare has made interaction with medical staff easier and more efficient by improving the connectivity of healthcare-related devices [5]. On the one hand, IoT technologies play an important role in improving the technological areas but on the other hand, the broad evolving nature of IoT with different embedded devices has opened a road towards advanced security challenges.

IoT systems have a large attack surface due to the internet-supported connectivity of IoT devices. The intruder accesses confidential information from a communication channel by eavesdropping. DoS, Probe, U2R, and R2L regarded as encapsulated forms of attacks that take place in the whole network and intended to disrupt the environment.

In IoT systems development and implementation of multiple defense mechanisms protect the information from attacks. The intrusion detection mechanism detects attacks or unauthorized access by analyzing system activity in IoT as shown in Figure 1.1. An IDS is a powerful security system, which is used to protect the IoT embedded environment by maintaining adequate network protection [6][7]. IDS categorizes according to implementation and detection methodology. IDS can be categorized as host based IDS or network based IDS based on implementation whereas according to detection methodology it can be categorized as anomaly based IDS , signature based IDS, specification based IDS , or hybrid IDS [8][9].

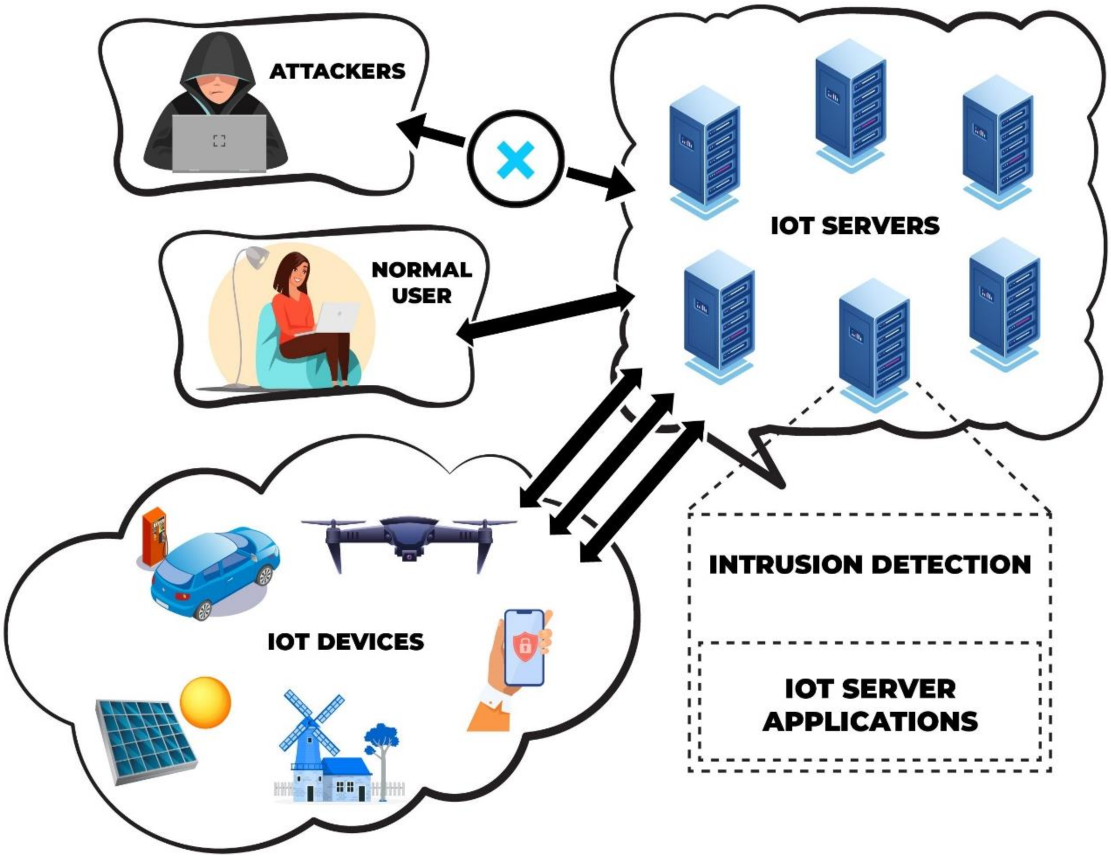


Figure 1.1: Intrusion detection in IoT

There are numerous techniques to detect the intrusion in an IoT system but two techniques called ML and DL are more efficient intrusion detection techniques. These techniques detect the intrusion at an early stage in an IoT system. ML-based approaches have been widely used for attack detection to help the network administrator by taking the proper measures to prevent intrusion into the network [10]. DL is a branch of ML and Artificial Intelligence(AI) which gain popularity among researchers due to two main attributes hierarchical feature representation and long-term temporal pattern dependencies learning [11]. DL approaches are suitable in such an environment where a large dataset is involved and become familiar with IoT systems. IoT-based applications produce a large amount of data. ML or DL-based IDS face challenges such as low detection rate and high false-positive rate. The development of hybrid IDS meets the solution to these challenges by combining the benefits of different approaches.

1.2 Motivation

Network connectivity, mobility, and communication have grown due to IoT. It became the reason for the increase of security attacks such as DoS, Man-in-the-middle attack (MiTM), hacking, interruption, etc. in IoT. According to a report [12], it is predicted that there was 776% growth in attacks between 100 Gbps and 400 Gbps from 2018 to 2019 and attacks will double to 15.4 million by 2023 globally as shown in Figure 1.2.

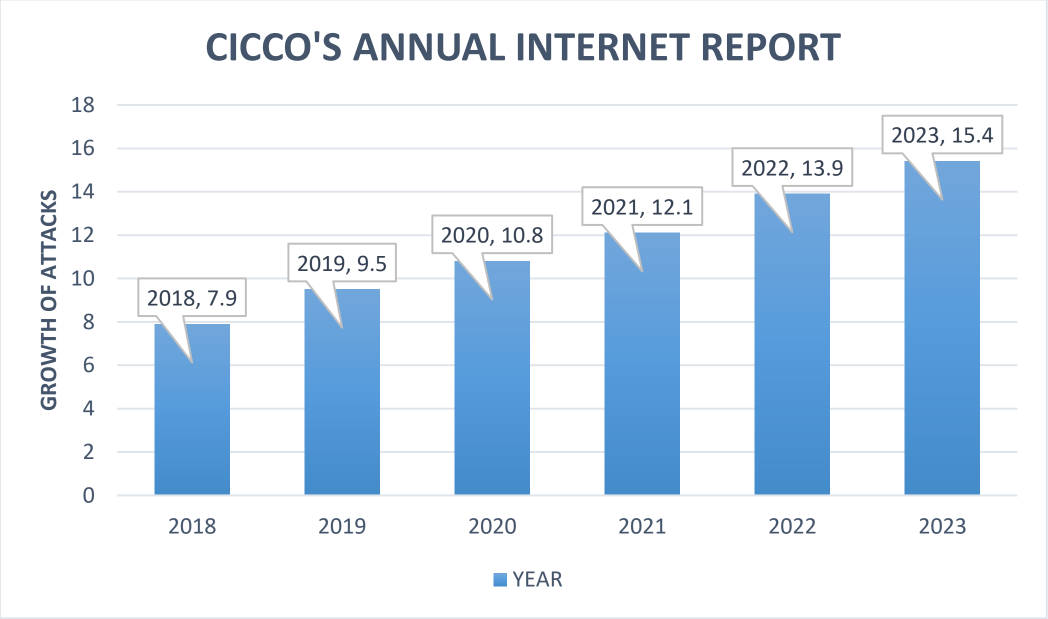


Figure 1.2: Growth of attacks according to Cisco’s Annual Internet Report (2018-2023)

Irregular update of IoT devices causes vulnerability of IoT systems. A single vulnerability can penetrate IoT devices and enter into the IoT network. Furthermore, it opens door to a big-scale attack. Intrusion detection in IoT proceeds through the development of ML and DL techniques [11]. Hybrid intrusion detection combines the benefits of combined approaches and overcomes the drawbacks of each approach [13]. Research based on the hybrid intrusion detection in IoT-based smart healthcare with better detection capabilities is necessary. Therefore considering the gap in the field of IoT our main concern is to develop an improved hybrid intrusion detection by using the latest dataset and improving the performance of intrusion detection.

1.3 Architecture of Intrusion detection in IoT

The IoT architecture consists of four layers such as perception layer, network layer, middle-ware and perception layer as shown in Figure 1.3.

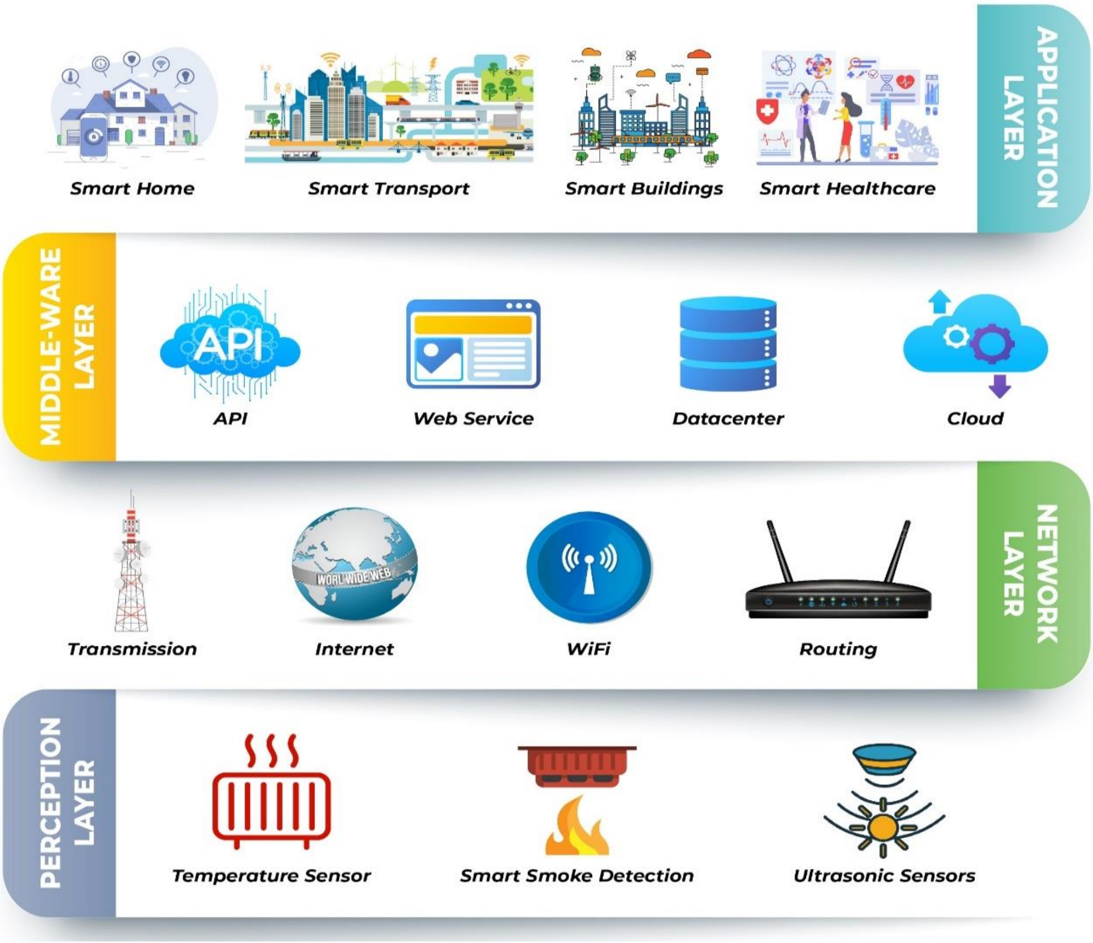


Figure 1.3: Architecture of Intrusion detection in IoT

The first layer is perception layer, which consists of actuators, sensors such as RFID, WSN, blood pressure sensors, temperature sensors, motion sensors etc. The next layer is network layer, which based on the interlinking devices that share the information with upmost layer through protocol. The next layer is the middle-ware layer that contains servers, GPU and

APIs that stores and process the information, which works as middle-ware between network layer and upper layer. The top layer is application layer, which provides services to various users based on their needs.

1.3.1 Perception Layer

The first layer of IoT architecture is perception layer, which deals with sensors. Different sensors collect information. The type of information depends on sensor i.e. temperature, ultrasonic, smart smoke detection, location, direction, movement etc. By using the devices in this layer, the collected information have sent and received to and from the upper layers. It ensures the security of communication devices.

1.3.2 Network Layer

This layer focuses on transmission of information and network access provision to internet. Therefore, the collected information from sensors in the perception layer transmits through network layer. Different communication technologies including 3G, 4G, 5G, GSM, Bluetooth, Wi-Fi, IPv6, etc. are used in this layer for transmitting information. As this layer is a complex and vulnerable part in IoT architecture leads to different security attacks such as MiTM, DoS, Eavesdropping/Sniffing and routing attacks etc. Intrusion detection has performed at this layer.

1.3.3 Middle-ware Layer

The middle-ware layer based on local clouds, API servers etc. is used to store and process the information. The needs of application layer are fulfilled through APIs provided by this layer. Moreover, security of database and cloud are the rest of the security challenges in this layer.

1.3.4 Application Layer

This layer has an intelligent detection system, which fulfills the needs of various applications of IoT like healthcare, smart cities and smart buildings etc. and provides many services to users. Data privacy, reliability, and authentication of confidential information from customers are security issues that need to be taken care at this layer.

1.4 Applications of IoT empowered intrusion detection

IoT empowered applications provide hand-to-hand solutions to the users by integrating the medical devices with IoT and formulate Internet of Medical Things (IoMT).Intrusion detection is performed to protect the IoMT environment [14]. Intrusion detection have also gained popularity in IoT based drones due to integrating technologies such as sensors, cameras, transmitters in Network of drones(NoD) [15].

1.5 Constraints of detecting intrusion in IoT

In the following section, various constraints including large-scale attacks, IoT security related dataset availability, computational complexity, bad quality of data, irrelevant features in ML/DL based intrusion detection models, poor performance of trained model, high false alarm rate and low detection rate, model overfitting, model underfitting, model generalization and detecting intrusion in real-time environment.

1.5.1 Large-scale attacks

IoT is a system based on the interconnected environment. This interconnection of devices are becoming the reason of massive growth of IoT devices. This network is not only expanding the surface of attack but also the magnitude of the attack. The attackers can launch large-scale attacks that are not easy to control. Botnets such as Mirai botnet and Distributed Denial of Service (DDoS) is an example of large-scale attacks. Developing intrusion detection mechanism for fencing off these attacks is limitation in IoT systems [16].

1.5.2 IoT security related dataset availability

The rapid increase of IoT devices produces massive attack surface for intruders to introduce advanced attacks. Intrusion detection data selection plays important role in intrusion detection systems [11]. The old traditional intrusion detection datasets are limited and do not tackle the latest, unknown and real world attacks. Therefore, IoT related dataset based on IoT devices network traffic with continuously updating latest attacks is one of the big challenge.

1.5.3 Computational complexity

IoT devices are resource-constrained devices. IoT devices resources are categorized as energy, memory, processing etc. [17]. These resources are essential for ML and DL deployment but are limited and make the real time implementation difficult [18]. There is a need to develop ML and DL based approaches that can reduce computational complexity. Development of real time detection and prevention system are important for effective security mechanism specifically for large-scale IoT systems. So computational complexity reduction is practically important in IoT.

1.5.4 Bad quality of data

Quality of data is very important for accurate outcomes. Model training without analyzing model is a bad approach. It leads to noisy data that limit the effectiveness of intrusion detection system. Data preprocessing improves the quality of data by removing the outliers and filtering missing values. Furthermore, model training by labeled dataset helps to build such learning models that provide the basis of future data prediction [19].

1.5.5 Irrelevant features in ML/DL based intrusion detection models

The large number of irrelevant features in training data leads to unexpected outcomes. Therefore, researchers perform feature selection to improve the accuracy of model [20][21] .The accuracy of selected features effects the performance of model [22].

1.5.6 Poor performance of trained model

Performance evaluation helps to measure the performance of trained model. Poor performance of attack detection using ML and DL leads to low efficiency. Relative lack of training data often leads to poor performance [23]. The good performance shows that unknown malicious activities in IoT systems detected in an efficient way by using ML/DL models.

1.5.7 High False alarm rate and low detection rate

High false alarm rate often results due to large volume of data [24]. This is one of the critical challenges in case of unknown attacks. False alarm rate is the probability of false detection. A model with high probability of false intrusion detection is not considered reliable for the securing the IoT networks. Higher percentage of false alarm rate leads to low detection rate.

1.5.8 Model Overfitting

Overfitting occurs when a model performs very well on training data but it shows poor results on test data. Low bias and high variance leads to model overfitting [25]. Moreover, too much training time makes the model more disposed to overfitting.

1.5.9 Model Underfitting

Underfitting occurs when training error is high. High bias and low variance becomes the reason of model underfitting [25]. If a model is underfit, it is not reliable enough for detecting attacks in IoT networks in efficient way.

1.5.10 Model Generalization

The model that is free from underfitting and overfitting performs well on ML [25]. If a model performs well on training data and shows good results on testing data then it is a generalized model. The lack of generalizability in model shows that it does not have the ability to detect attacks in IoT environments.

1.5.11 Detecting intrusion in real-time IoT environment

Implementation of intrusion detection model in real-time IoT environment is a big challenge. Performance evaluation of identification of intrusion in real environment shows the real capability of model. Despite providing tremendous results of intrusion identification, the research [26] hardly implemented in real-time environment.

1.6 Problem Background

IoT proliferation has enhanced lives of people by interconnecting smart devices and applications in almost every domain of life. The increase in use of IoT based applications opened a challenge of security threats in IoT environments. Many existing researches have conducted for detecting threats in IoT based networks using ML and DL. It gained popularity and still improvement is desirable for securing IoT environment. Many traditional Intrusion detection techniques have used for identifying the threats in IoT environment but these techniques are not secure enough for IoT networks. ML technique known as RF detected the intrusion with low classification error and improved accuracy but did not works well in real time monitoring in case of different trees. With the rapid increase in technology, attackers are launching latest attacks i.e. DoS, DDoS, Mirai, BotenaGo etc. to breach the security of IoT systems. DT technique worked well in case of known attacks but not ideal for detecting unknown attacks. In case of extensive network traffic, SVM performed poor. Moreover, PCA detected the intrusion briskly and reduced the complexity of data in IoT based smart systems but it lost the important information of system.

DL based intrusion detection is carried out by many researchers to reduce the training time and improve the accuracy of model. MLP based techniques reduced the feature dimensionality reduction but accuracy did not improve. RNN techniques detected intrusion in many networks but it has the limitation of long-term dependency and losing the information at any time interval, which affected the performance. Some schemes got the better results but with the errors of overfitting and underfitting.

However, the attention was rarely given towards overfitting and underfitting, real-time monitoring, latest IoT dataset, data normalization, feature selection. Moreover, the good performance of model shows the capability of detecting intrusion in a better way. To fill the gap, this study presents hybrid ML and DL based intrusion detection scheme by using latest IoT related dataset. It can detect the intrusion with good performance by model generalization, dealing with undrefitting and overfitting.

1.7 Problem Statement

The lack of IoT security leads to vulnerabilities, which allow the launch of various attacks i.e. Mirai, Botnet, DoS, DDoS , MiTM etc. through IoT [27]. The traditional intrusion detection technologies cannot handle intrusion and malicious activities in the complex environment of IoT [28]. The traditional intrusion detection datasets lack advanced attacks and features of real time traffic. The development of hybrid ML and DL based intrusion detection mechanism by using the latest dataset often does not provide the correct measure of effectiveness.

1.8 Research Questions

The study presents the following questions:

i. How to develop hybrid intrusion detection in IoT by using IoT related latest dataset?

ii. How to make the mechanism of hybrid intrusion detection in IoT effective?

1.9 Aim of Research

The growth and diversity of interconnected devices in an IoT system making it vulnerable to latest attacks. These attacks launched by cybercriminals or hackers can steal the important information of an IoT network. It is very important to monitor the malicious activity continuously in order to protect the IoT environment. For detecting the intrusion in such and environment learning model with good performance is the basic need. The aim of the study is to develop a hybrid intrusion detection in IoT by using latest dataset with IoT traces that can detect the attacks in IoT environment effectively. To achieve this goal firstly this study reviews the state of art machine learning, deep learning and hybrid models. This study also examines the performance evaluation of these models by using various datasets. Finally, a solution consists of utilizing the strengths of hybrid models in IoT by using latest IoT related dataset. To make the mechanism of proposed approach effective this study uses various performance metrics and presents the performance comparison with various existing models.

1.10 Research Objectives

i. To develop the hybrid intrusion detection model in IoT by using IoT related latest dataset.

ii. To make the mechanism of hybrid intrusion detection in IoT effective.

1.11 Scope of Research

The scope of this study is to develop effective intrusion detection system in IoT based smart healthcare environment by using proposed hybrid model. Moreover, it is suitable for real time smart environment.

1.12 Thesis Organization

The thesis organization is as follows

Chapter 2 will present the detailed state of art ML, DL and hybrid schemes review for detecting intrusion in IoT. It includes detailed overview of all the existing schemes and discusses how this study differentiate itself with the existing schemes. Furthermore, it includes a detailed comparative analysis of state of art schemes based on ML, DL and hybrid intrusion detection in IoT in the form of table that give direction towards new research.

Chapter 3 will present the methodology and detailed interpretation of identified problem solution. It presents the proper planning, research design and development of proposed scheme. Experimental design is setup for evaluating the performance of proposed schemes.

Chapter 4 will present about analysis and results of proposed scheme. It will present the performance evaluation of hybrid model in the form of graphs and tables. Furthermore, it will also presents the comparison of benchmark schemes with proposed model.

Chapter 5 will sum up the contribution of this research work. It will also presents the gaps of proposed model, which lead toward further directions for future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Overview

In this chapter, a detailed overview of schemes for detecting intrusion in IoT discussed. Intrusion detection in the IoT domain is a big challenge because these IoT devices have endangered various types of attacks. This chapter includes a detailed discussion and taxonomy of intrusion detection schemes in IoT. Moreover, the schemes categorized under different headings of Machine Learning (ML) based intrusion detection, Deep Learning (DL) based intrusion detection, and hybridization of both Machine Learning and Deep Learning-based intrusion detection. Moreover, the chapter includes a discussion about the literature review to highlight the strengths and shortcomings of each study. Finally, research challenges also highlighted.

2.2 Schemes for detecting intrusion in IoT

Various studies dealing with intrusion detection in IoT using DL, ML and hybrid schemes have discussed in the literature. Figure 2.1 presents the taxonomy for intrusion detection schemes in IoT.

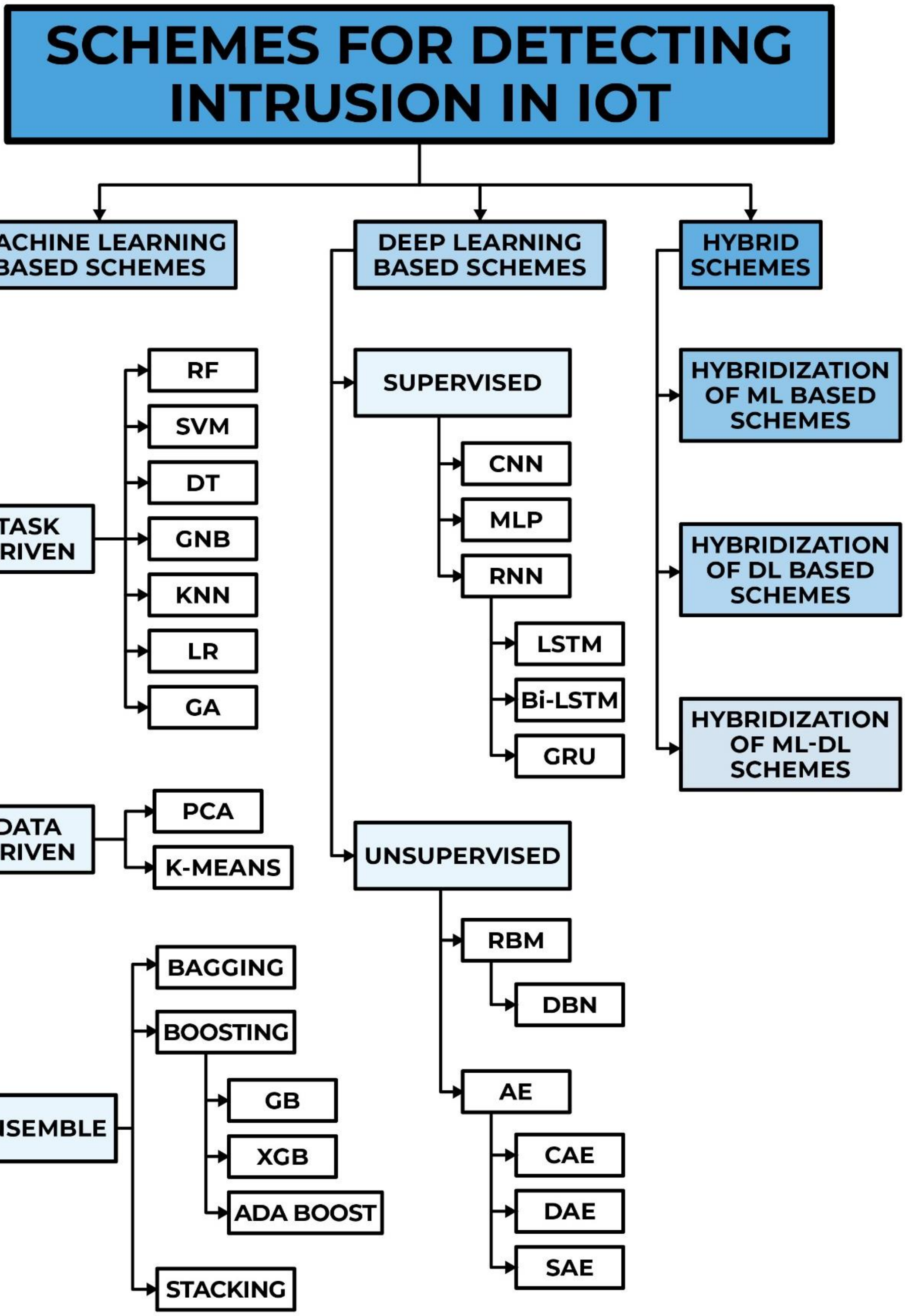

SCHEMES FOR DETECTING INTRUSION IN IOT
MACHINE LEARNING BASED SCHEMES
DEEP LEARNING BASED SCHEMES
HYBRID SCHEMES
TASK DRIVEN
RF
SVM
DT
GNB
KNN
LR
GA
DATA DRIVEN
PCA
K-MEANS
ENSEMBLE
BAGGING
BOOSTING
GB
XGB
ADA BOOST
STACKING
SUPERVISED
CNN
MLP
RNN
LSTM
Bi-LSTM
GRU
UNSUPERVISED
RBM
DBN
AE
CAE
DAE
SAE
HYBRIDIZATION OF ML BASED SCHEMES
HYBRIDIZATION OF DL BASED SCHEMES
HYBRIDIZATION OF ML-DL SCHEMES

Figure 2.1: Taxonomy for Detecting Intrusion in IoT

2.2.1 Machine learning based Schemes for Detection Intrusion in IoT

This section includes some of the studies on different ML-based schemes for detecting intrusion in IoT networks. The two most widely used Machine Learning techniques are Task Driven and Data Driven. Task Driven techniques are Supervised Machine learning techniques whereas Data Driven techniques are unsupervised Machine learning techniques. Different task-driven schemes namely Random Forest (RF), Support Vector Machine (SVM), Decision Tree (DT), Gaussian Naïve Bayes (GNB), K Nearest Neighbor (KNN), Logistic Regression (LR), Genetic Algorithm (GA), and Data-Driven schemes namely Principal Component Analysis (PCA) and K- Means Clustering for detecting intrusion in IoT discussed in the literature:

Random Forest (RF), a compound of tree structure classifiers is commonly used on intrusion detection data for classification and regression analysis [29]. RF has the characteristic of achieving high classification accuracy. RF classifier was used by Farnaaz and Jabbar [30] for intrusion detection. RF has applied to the NSL-KDD dataset to deal with four attacks: DOS, probe, U2R, and R2L. Results in the form of low false alarm rate and high detection rate proved that RF increases the accuracy of classification. RF has low classification error as compared to other traditional classification techniques and enhances the accuracy of classification. The main drawback of the RF algorithm is that the algorithm becomes slow for real-time forecasting in the case of numerous trees. The authors of [14] used RF with Particle Swarm Organization(PSO) for intrusion detection in the Internet of Medical Things (IoMT) based smart environment. In this scheme, the performance of the proposed scheme compared with other approaches, and the proposed model performed well, however, there is stillroom for improvement in terms of F1 score.

To overcome the problems of intrusion detection for unseen data and the rise of false-positive rates Hanif et al. [31] presented Artificial Neural Network (ANN) based IDS.UNSW-15 dataset is used for performance measurement which is based on benign network data and different types of malicious and diversified data. IoT controller applied in the proposed scheme used to classify non-benign data and drop whenever an attack occurs. The proposed approach achieved 84% accuracy and a low false-positive rate. However, there is still a need to improve the performance of the proposed model for intrusion detection. Moreover, combining the proposed approach with real-time network traffic is one of the big challenges of this study. The authors of [32] proposed a machine learning-based Intrusion Detection System (IDS) which is applied in IoT as a service. In this scheme, Random Forest (RF) used as a classifier for intrusion detection, and Artificial Neural Network (ANN) is used for the classification of detected intrusion. UNSW-NB15 [33] dataset is used for performance evaluation of proposed model. The proposed model performed well in intrusion detection, but detected intrusion classification lacks good accuracy.

The researchers have been influenced to apply different classifiers such as Support Vector Machine (SVM) in Intrusion Detection System (IDS) to deal with rapidly growing security attacks[34]. In [35] the authors presented SVM based intrusion detection system with augmented features and declared their model highly competitive for intrusion detection system as compared to other schemes. Logarithm Marginal Density Ratios Transformation (LMDRT) has implemented to create the original features to obtain new and improved features. However, they did not show the number of training and testing samples and other statistics of the dataset. Moreover, a wide range of data results in performance reduction of SVM, and it has not considered the better option to cope with massive network traffic for intrusion detection. Jing et al. [36] proposed SVM-based intrusion detection with binary and multi-classification experiments. The SVM model applied with a nonlinear scaling method to cope with the traditional normalization limitation of depending on minimum and maximum values of sample data. The performance evaluated using the UNSW-NB15 dataset and compared with DT, LR, NB, ANN, and Expectation-Maximization clustering models. Although the proposed models performed comparatively better than other models still improvement is required for intrusion detection.

DT is considered suitable for familiar intrusion techniques because it results in the form of better detection accuracy for familiar intrusion techniques; however, for unknown intrusion techniques it is not considered an ideal model[22]. For effective intrusion detection, an IDS is proposed in [37] by using DT. Prediction performance of the proposed model has improved by using correlation feature selection (CFS). The feature selection approach has applied separately to binary and multiclass classifications. Furthermore, performance evaluated separately for both categories. However, from the experimental outcomes, the method did not clear; there is a scope for enhancement. The authors in [4] proposed a decision tree based model to detect intrusions in IoT based smart grid in an efficient way. This model has based on the combination of three decision trees. This combination-based model has used for classification. It showed that this scheme is beneficial for intrusion detection system in IoT-based smart grid, as the results have compared with other techniques. However, the proposed scheme's recall value is less than other techniques.

Gaussian Naïve Bayes (GNB) or Naïve Bayes (NB) is statistical and a supervised learning technique for classification. For composing this technique, the training dataset is used to assess the probability of every class considering the feature value of new instances [38]. Naïve Bayes is used in many studies [39][40][41] as a performance comparison of various models. Moreover, NB is combined with improved PCA for detecting the latest types of attacks [40].

KNN based classification has been used by some researchers generally for intrusion detection in [42] and specifically in IoT-based network intrusion detection in [43][44]. Swarna Sugi et al. [43] proposed one ML based intrusion detection by using KNN and another DL based intrusion detection by using LSTM.The performance of both models has evaluated and compared by using the Bot-IoT dataset. However, KNN did not show efficiency in attack detection accuracy as compared to LSTM. Another study based on Machine learning algorithms is presented in [39]. In this scheme, the main attention given to DDOS attack detection for improving enterprise network security. The performance of the two ML-based models: KNN and NB has analyzed by using two datasets. The experiments executed on binary classification. KNN performed well than Naive-Bayes with higher detection accuracy and low error rate. However, the model’s performance has not evaluated for multiclass classification.

For the development of IDS, Biswas analyzed different feature selection techniques and machine learning classifiers in a comparative study [41]. Four feature selection techniques namely Information Gain Ratio (IGR), Correlation-based Feature Selection (CFS), Minimum Redundancy Maximum Relevance, and Principal Component Analysis (PCA) have presented in this study. Furthermore, various machine learning classifiers have discussed in this study. Due to the pros and cons of all of the feature selection techniques and machine learning classifiers, the author has used different combinations of feature selection techniques and machine learning classifiers. To find the results, five-fold cross-validation has applied to the NSL-KDD dataset. KNN classifier performed better than other classifiers with higher accuracy and all the combinations of KNN and IGR feature selection achieved the highest accuracy. Anomaly-based IDS is developed for IoT network intrusion by applying various ML algorithms by Liu et al. [16]. By labeling anomalies as zero and normal packets as one, binary classification applied. Moreover, three trials has performed for each algorithm. The performance has evaluated by using the Network Intrusion dataset. KNN and XGBoost achieved good results in terms of accuracy and other measures. RF achieved the highest metric scores but it carried out excessive computational effort. SVM performed poorly despite consuming abundant computational resources. Despite consuming less computational resources, LR performed satisfactorily because data is not as normalized as required to overcome poor accuracy.

Logistic Regression is useful for classification. Assessment of the probability that the instance belongs to a specific class is being done through Logistic Regression [45].To handle attacks and anomalies such as Denial of Service (DoS), Malicious Control, Data Type Probing, Malicious Operation, Spying, Scan, and wrong setups that can badly affect IoT systems the authors in [46] compared various ML techniques. The Logistic Regression (LR) has examined for various types of attacks and anomaly detection. In terms of accuracy DT, RF and ANN achieved the same accuracy, which was higher than other ML techniques. However, RF outperformed the other techniques in terms of other metrics.

In the many recent years, different ML based schemes have been widely used to make the security of IoT systems more powerful. Principal Component Analysis (PCA) has been used in various studies [40, 41, 47] for detecting intrusion. Principal Component Analysis (PCA) is considered efficient due to rapidly detecting intrusion behavior, compressing data complexity, and determining the most important features, however, it has a limitation of losing important information [40]. By reducing the dimensions of data, PCA improved the quality of data [47]. Moreover, when PCA is combined with GNB in [40], it results in reducing the detecting intrusion time.

Prachi Shukla [48] proposed three Machine Learning based intrusion detection systems for Wormhole attack detection in IoT. K Means clustering an unsupervised learning algorithm named KM-IDS used for intrusion detection in IoT. Another ML-based technique named Decision tree implemented for intrusion detection systems in IoT. Furthermore, hybridization done by combining both ML techniques KM-IDS and DT-IDS to propose a new ML-based ID for IoT. The experimental results showed that K-Means-based IDS achieved a higher detection rate than the other two proposed approaches. However, K Means-based IDS got a high false-positive rate and the author declared hybrid IDS more accurate for IoT with a lower false-positive rate than the other two proposed approaches.

A genetic algorithm is a heuristic technique used to solve optimization problems. Genetic Algorithm based IDS of IoT applications is proposed by Jain et al. [49] in IDS. The proposed model has tested using KDD Cup 99 dataset. Selection, crossover, and mutation operations performed on the dataset and the Genetic algorithm declared a useful algorithm for the security of IoT-based applications. DT and GA-based IDS has been presented in [50]. Primary solutions optimized through natural evaluation process by using GA as shown in Figure 2.2.

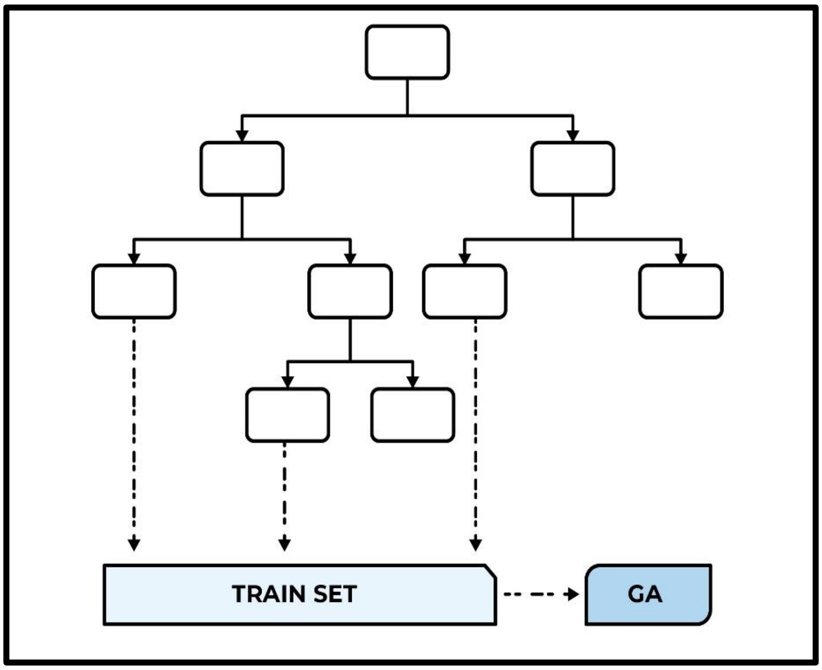


Figure 2.2: GA and DT based IDS [50]

Ensemble learning can enhance the weak classifiers to generate improved results so this ability is considered better than a single classifier [51]. Boosting, Bagging, and Stacking are various ML-based methods, which have been proposed by different researchers [13][51]. Weak learners transform into strong learners through boosting. Weak learners learn from earlier misclassifications and develop a stronger learning model through the ensemble Gradient Boosting algorithm [52]. Extreme Gradient Boosting (XGBoost) is the boosting algorithm that has been applied in many studies [16, 53]. XGBoost is scalable machine learning system that handles the problems of overfitting and improve the model generalization ability [54]. Another boosting classifier known as Ada Boost (AB) is flexible meta-estimator that improves the performance of model by learning through giving the preliminary training weights on actual dataset [54].

Khraisat et al. [13] proposed the hybrid IDS by stacking ensemble of DT based classifier C5.0 with OC-SVM. This model combines the characteristics of both AIDS and SIDS. This scheme detected both common intrusions and zero-day attacks with low false-alarm rates and high detection accuracy.

2.2.2 Deep Learning-based Schemes for Detecting Intrusion in IoT

Various schemes have presented for Detecting Intrusion in IoT in this category. Deep learning is the modern technique used to improve the security of IoT networks. Deep learning-based schemes have categorized as Supervised, Unsupervised, and Hybrid. In the following Supervised DL schemes such as CNN, RNN and Unsupervised deep learning schemes such as AE, DBN, BM, and SPN discussed:

Huong et al. [55] proposed a new deep Learning-based Convolutional Neural Network(CNN) for detecting intrusion in IoT systems. Making a CNN-based general model for detecting intrusion in IoT is one of the main contributions of this study. The proposed model has applied in three phases: a collection of data, preprocessing, training of the network, and detection. The performance of the proposed scheme achieved higher accuracy in comparison to other approaches. However, only a limited number of classes and samples have used in this scheme.

Rapidly increasing botnet attacks are leading to false alerts and low recognition precision. To handle different types of Botnet attacks, the authors in [56] proposed IDS for IoT using four deep learning models: CNN, Simple RNN, LSTM, and GRU. The performance of the proposed models evaluated and compared using the BoT-IoT dataset. CNN-based IDS has proved efficient by victoriously detecting various types of attacks. The CNN-based model achieved higher accuracy in contrast with other model. However, the model has not implemented in real network scenario.

Multilayer Perceptron(MLP) is used for feature dimensionality reduction after feature selection [57]. For DoS attack detection RF as well as MLP is applied by Wankhede et al. [58]. Packets have classified as benign or DoS attack in the proposed scheme. RF achieved higher accuracy than MLP. However, reducing the number of features is a big challenge of this research.

Another DL based IDS is proposed by Khan et al. [59]. using DNN for detecting attacks in IoT network. Two datasets namely MQTT-IoT-IDS 2020 and three types of attacks Denial of Services (DoS), intrusion in the network, and Man in the Middle (MitM) based dataset [60] are used for performance evaluation . Furthermore, different ML-based techniques, namely KNN, RF, NB, DT, LSTM, and GRUs compared with the proposed modelFor the MQTTIOT-IDS 2020 dataset, the proposed model achieved higher accuracy for binary classification whereas for multi-label classification the accuracies are less. However, developing a deep learning-based model for advanced vulnerabilities on different IoT protocols is the limitation of the study. Four DL models namely DNN, MLP, CNN, and AE are presented in [61] for detecting malicious activities in IoT networks. DNN performed well among all models with higher accuracy.

Yin et al. [10] presented RNN-based IDS for detecting intrusion using deep learning. The directional loop of RNN used to remember the prior information and it has applied to the present output. The performance of the proposed model studied in binary classification and multi-classification. The performance of RNN-IDS compared with RF, SVM, ANN, J48, and other machine learning schemes. RNN-IDS has proved a suitable model for classification.

The Long-Short Term Memory (LSTM) is an improved RNN that solves the traditional RNN's issue of long-term dependence and does not forget the important information at any time interval [62]. To handle the challenge of detecting large scale attacks, the authors proposed bidirectional LSTM(BiDLSTM) [63]. The RNN's issue of vanishing gradient has resolved by this method. Two LSTMs have trained on input data; the first LSTM trained on original input data and the second one has trained on a duplicate of input data. Furthermore, k-fold cross-validation method used for performance validation. The result of the proposed model has compared with conventional LSTM. The proposed model achieved higher detection accuracy and lower false alarm rate than other models. However, the limitations of the study bring up in the form of higher complexity and run-time analysis of the BiDLSTM model. There is a need to increase the training time than other models. GRU and LSTM have an identical design but the characteristic of GRU, which makes it different from LSTM is supervising the forget factor and decision to update the state unit [64]. GRU has trained easily in contrast to LSTM and training efficiency enhanced.

Autoencoders (AEs) are multilayered neural networks that are capable of learning by self-supervision to reorganize data [65]. For detecting network anomalies, the Autoencoder-based method is proposed by Chen et al. [66]. Nonlinear correlations among features have captured by the autoencoder. Dimensionality reduction has considered one of the trendy methods for detecting anomalies. Convolutional Autoencoder (CAE) has applied for dimensionality reduction to shorten the training time. It consists of convolutional and deconvolution layers. The convolutional layer used in the encoder part and the deconvolution layer in the decoder part. The proposed method has compared with different methods based on network anomaly detection. The performance has evaluated by using the NSL-KDD dataset and the proposed method performed well in comparison to other methods. Another type of autoencoder is Denoising Autoencoder (DAE). Attacks-related features have given extra attention by applying the weighted loss function. These feature plays an important role the in enhancement of the performance of intrusion detection [57].

Li et al. [67] proposed an improved Sparse Autoencoder (SAE) as a classification model to enhance the detection rate of the small proportion of the model's attack data. Softmax classifier has combined with SAE to improve the performance of the proposed model. Continuous adjustment of parameters has applied for reducing the false positive rate (FPR). The performance of the proposed model has compared with RF, SVM, and NB by using the NSW-NB15 dataset. The overall accuracy and detection rate of the proposed model are higher but there is stillroom for improvement in terms of accuracy. However, for individual data labeling RF has higher detection effect than the proposed model.

RBM is an unsupervised deep learning model in which pattern extraction is done in an unsupervised way and is considered suitable for unsupervised feature learning [68]. Dawoud et al. [69] proposed Software Defined Network (SDN) based IoT architecture. The bottom layer has based on IoT devices whereas the SDN layers: the forward layer and controller layers top the IoT devices. The RBM-based anomaly detection system has developed in this scheme. The RBM-based IDS has proposed at the controller layer making possible the direct communication of IDS with the network. The proposed scheme's performance is assessed by using KDDCup '99 dataset. This anomaly detection scheme attained 94% precision, which is better than SVM and PCA. However, there is stillroom for performance improvement in the proposed model. Furthermore, they did not represent the training and testing samples of data. Practical implementation of the proposed scheme is still a big challenge. RBM is an elementary unit for DBN and DBM. In [28] DBN is used in combination with an improved Genetic Algorithm to detect the intrusion in the IoT network. A Deep Belief Network (DBN) based on multiple RBMs and all RBMs trained individually.

2.2.3 Hybrid Schemes for Detecting Intrusion in IoT

Various studies [32] [40] [47] [70] are based on hybrid Machine Learning Schemes by integrating the machine learning techniques for efficiently detecting intrusion. In [40] Zhang et al. combined Gaussian Naïve Bayes (GNB) with improved PCA for detecting the latest types of attacks in less time. PCA improved the dimensionality reduction of data and the GNB classifier detected the intrusion behaviors. Binary classification based on two categories normal and abnormal is used. The proposed approach resulted in the form of higher accuracy detection and less intrusion detection time as compared to traditional machine learning schemes. Due to the categorization of classes as normal and abnormal, this study did not cover the effects of detection on different types of attacks and not satisfactory for a specific type of attack. Principal Component Analysis (PCA) is used with a random forest classification algorithm in [47]. In this study, PCA has used to enhance the quality of data by reducing the dimension of data whereas Random Forest has used to classify data. Results have compared with other ML-based techniques such as Naïve Bayes, SVM, and Decision Tree, and the proposed scheme performed well in terms of higher accuracy as compared to other techniques. In [70] the authors proposed 2-class Support Vector mechanism and decision tree to implement such an intrusion detection system which provides more effective detection rate than a single SVM. They decomposed the network packets into four parts: icmp, udp, tcp and application layer. As compared to traditional single SVM the proposed model consumed very less training time, however, attack-detecting accuracy is poor.

A Hybrid DL-based IDS named HCRNNIDS is proposed by Muhammad Ashfaq Khan et al. [11] to handle the problems of detecting intrusion on a big data processing architecture. A hybrid Convolutional Recurrent Neural Network has constructed by introducing recurrent layers after CNN layers. Local features captured by CNN through convolution whereas temporal features captured by RNN to enhance the performance of IDS. The performance of the proposed scheme has evaluated by using CSE-CIC IDS 2018 dataset. The results showed that the proposed model proved to be more efficient in contrast with other approaches. Difficulties in computational problems of intrusion detection have decreased with the help of the proposed scheme. Furthermore, the scheme has not tested on different datasets it will not work efficiently for changed network traffic.

Four deep learning models: MLP, CNN, LSTM, and a hybrid of CNN+LSTM are proposed by Roopak et al. [71] for DDoS attacks identification in IoT networks. Sigmoid activation function has applied in a dense layer that is the last layer in all models. CICIDS2017 dataset used for the performance evaluation of proposed models. Furthermore, the performance of proposed models compared with machine learning methods: SVM, GNB, and RF. The hybrid model attained better performance in terms of higher accuracy. Another hybrid model CNNGRU model is proposed by Ullah et al. [27] to detect anomaly-based intrusion in IoT networks. Different attacks have classified in multiclass classification by separating them from normal network traffic. The binary class proposed model is based on an input layer, a convolutional layer, a GRU layer, a flatten layer, a fully connected dense layer, and an output layer whereas the multiclass proposed model has all the same number of layers as in binary class model except two convolutional layers and two GRU layers. Three IoT related datasets have for performance evaluation of proposed model. The CNNGRU binary and multiclass classification models yielded good results in comparison with CNN, LSTM, and Feed Forward Neural Network (FFNN).

A variant of Autoencoder called LSTM-AE is presented by Xu et al. [62] for intrusion detection. In the proposed scheme, data’s time series features are processed by LSTM, and an autoencoder is used for intrusion detection through feature learning ability as shown in Figure 2.3. Just like an autoencoder, it consists of an encoder and decoder but an LSTM unit replaces the neuron. The input time series attribute data's hidden state expressions have learned by the encoder and the decoder has used to reorganize the data in converse order by using hidden state expressions. Moreover, five datasets with attack data including Fuzzing, Mirai, ARP, SSDP Flood, and Video Injection published by the Mirsky team used for measuring the performance of the proposed scheme. The used datasets consist of both pcap data and label data. After comparing the performance of LSTM-AE with other classification models, it has found that the proposed model achieved better accuracy and better F1 score. However, the performance decreases in the case of the number of negative samples in the dataset.

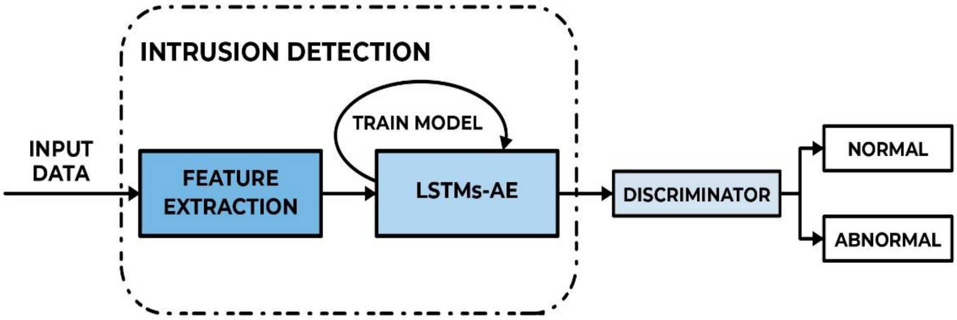


Figure 2.3: LSTM-AE based intrusion detection [62]

DL-based IDS is proposed in [57] for network security. Denoising Autoencoder (DAE) reduced feature dimensionality by selecting the limited number of important features with a weighted loss function. Then classification of selected data has performed through Multilayer Perceptron (MLP). The proposed model achieved satisfactory detection performance with a little feature selection ratio of 5.9%. However, there is still a need to deploy the proposed model in high-speed networks practically.

In [28] improved Genetic Algorithm (GA) is combined with Deep Belief Network (DBN) to detect the intrusion in the IoT network. To generate an optimal network structure multiple iterations have performed by GA and to classify the attacks this structure has applied as an intrusion detection model by DBN. From experimental results, the proposed model has proved to achieve higher accuracy than other models for detecting a specific type of attack. Moreover, the proposed model reduced the complexity of neural network structure.

2.3 Comparison of Intrusion Detection Schemes in IoT

Different intrusion detection schemes in IoT reviewed here to identify the gaps in the current research. In this section, comparative analyses of various schemes based on taxonomy have presented in tabular form, as shown in Table 2.1. The schemes analyzed based on the model, method, dataset, attack types, classification, performance metrics, benefits, and drawbacks. The drawbacks in different schemes give rise to possible gaps.

Table 2.1: Comparison of Intrusion detection schemes in IoT

Sr #	Ref	Model	Method	Dataset	Attacks	Classification	Performance Metrics	Benefits	Drawbacks
1	[30] 2016	RF	Network intrusion detection Classification of attacks Best Feature Subset Selection	NSL-KDD	Dos, Probe, R2L, U2R	Binary	Accuracy, Detection Rate, Mathews Correlation Coefficient, FAR	Low false alarm rate High detection rate	Need for multi-classification The model is not generalized Not suitable for real-time application Precision, Recall, and F1 Score are missing
2	[32] 2018	RF, NN	Intrusion detection in IoT as a service Feature extraction and classification of extracted features	UNSW-NB15		Binary and Multiclass classification	Precision, Recall, F1 Score	Intrusion is detected effectively	Underfitting Data Preprocessing and Feature Selection are not presented The performance of proposed models for detecting the

			by RF and NN						intrusion is not evaluated
3	[35] 2017	LMDRT-SVM	Network intrusion detection framework based on SVM feature augmentation via LMDRT classifier	NSLKDD		Binary	Accuracy, DR, FAR	Robust Performance of the proposed model Fast training speed	Research is not generalized to include various attack types Training and testing samples and dataset statistics is not mentioned Not suitable for massive network traffic
4	[36] 2019	SVM	Data Preprocessing with nonlinear log function scaling	UNSW-NB15	Analysis, Backdoor, DOS, Exploits, Fuzzers, Generic, Reconnaissance, Shellcode, Worms	Binary and Multiclass classification	Accuracy, FAR, FPR, DR	Accuracy is higher FAR is lower than other models in the binary classification model Training and testing accuracy is higher and FPR is lower than other models	Recall, F1 Scores are missing Improvement in accuracy is required Need to reduce the FAR in the binary classification model
5	[37] 2018	DT	Correlation-based Feature Selection (CFS) for selecting optimal feature Random Sampling for pattern selection Pattern Classification	NSL-KDD	Dos, Probe, R2L, U2R	Binary and Multiclass classification	Accuracy, DR, FPR	Unbiased Training of model Feature selection reduced FPR and improved performance Achieved higher accuracy Time and space complexity is reduced	The model is not applied to real-time network

			using CART(Classification and Regression Tree)						
6	[4] 2020	Hybrid DT	Intrusion Detection of IoT-based smart grid Three decision trees combined for classification Class creation using CART DT	NSL-KDD	Dos, Probe, R2L, U2R	Binary	Accuracy, Precision, Recall, F1 Score	Enhanced performance of IDS in IoT based smart grid with improved accuracy	Models performance is not evaluated for multiclassification The recall value is lower than other models Need for the improvements in the F1 Score The proposed model is not generalized Much time is consumed in modeling the combination of multiple trees
7	[39] 2016	KNN, NB	DDOS attacks detection for enterprise network security	KDD Cup 99 NSL-KDD	U2R, R2L, Dos, Probing attacks	Binary	Accuracy, Precision, Recall, F-Measure, Sensitivity, Specificity, Efficiency, Error Rate, BCR	KNN performed well in contrast to NB. XGBoost is suitable for real-time intrusion detection.	The model's performance is not evaluated for multiclass classification Model is not generalized
8	[16] 2020	SVM,KNN,LR, RF, XGBoost	Anomaly detection on IoT network intrusion by	IoT Network intrusion dataset	Mirai, MITM, DoS, Scanning attacks	Binary	Accuracy, F1 Score, Precision	KNN and XGBoost performed well	The model's performance is not evaluated for multiclass classification

			applying five ML algorithms						SVM required many computational resources RF required the highest computational effort LR achieved poor accuracy because data is not normalized Except for XGBoost rest of the models are not suitable for a real-time environment
9	[53] 2020	LR, XGBoost, RF, SVM, KNN, Ensemble Learning with LR, RF & SVM	Intrusion detection by using ML models on low power IoT devices	UNSW-NB15	Fuzzers, DoS Exploits, , Analysis, Backdoors, Generic, Reconnaissance, Shellcode, and Worms	Binary and Multiclass classification	Accuracy, Recall, Precision, F1 Score, and run time	RF outperforms other models Overfitting of RF model to each particular attack type is reduced XGBoost performed well in binary classification KNN achieved 2nd best performance in multi-classification	Overfitting in all models except RF Training time is not reduced The model is not generalized
10	[49] 2020	Genetic Algorithm	GA is proposed for IDS in IoT applications	KDD99				Beneficial for securing IoT based applications	The Proposed algorithm is neither simulated nor implemented Performance is not evaluated

									The Complexity of the model is not checked
11	[40] 2018	Improved PCA + GNB	Network Intrusion detection PCA for dimensionality reduction Weight coefficient improved PCA GNB Classifier for detecting intrusion behavior Compared with KNN, SVM,GDB,GNB, PCA+GNB	KDD99	DoS attacks, unauthorized access from remote hosts, unauthorized local superuser privileged access, and port scanning	Binary	Accuracy, Recall, Precision, F1 Score	High accuracy and less detection time	Training and testing statistics is not mentioned Not be suitable for unknown attacks Performance improvement required Limited to binary classification Not suitable for a specific type of attack
12	[47] 2020	PCA+RF	Intrusion detection over the internet PCA is used for dimensionality reduction RF is used for	KDD dataset			Performance time, accuracy rate, and error rate	Reduced performance time and error rate Higher accuracy	Training and testing statistics are not mentioned Model is not generalized Model is not classified Not suitable for specific attack type

			classification						Not suitable real-time network
13	[70] 2018	2-Class SVM +DT	Collaborative and adaptive intrusion detection model (CAIDM) Environments-classes, agents, roles, groups, and objects (E-CARGO) model developed by DT and SVM	KDD Cup 99	U2R, R2L, Dos, Probing attacks	Multiclass classification	Accuracy and training time	Consumed less time in training Performed better than single SVM	Attack detecting accuracy is poor Only accuracy is not enough to evaluate the performance of the model
14	[56] 2021	CNN	Intrusion detection in IoT Compared the proposed model's performance with RNN,LSTM and GRU	Bot-IoT	Service scanning, OS fingerprinting, DDoS TCP, DDoS UDP, DDoS HTTP, DoS TCP, DoS UDP, DoS HTTP, key logging , data	Multiclass classification	accuracy training, loss training, accuracy validation, loss validation , training time	Achieved higher accuracy Lower loss rates Less prediction time	The proposed model is not applied in real network traffic data and

					exfiltration				
15	[58] 2018	MLP, RF	DoS attack detection at application layer	CIC IDS 2017	DoS attack	Binary class classification	Accuracy	RF achieved higher accuracy	Need to reduce number of features Multiclass classification is not used
16	Intrusion	DNN	Intrsuion Detection for MQTT-enabled IoT smart systems Performance compared with different models	MQTT-IoT-IDS2020 MQTT dataset	Scan_SU, Scan-A,Sparta, Bruteforce,DoS,MitM,Intrusion in the netwrok	Binary and Multiclass classification	Accuracy, Recall, Precision, F1 Score	Achieved better results	There is a still need for improvement of performance
17	[61] 2019	CNN,MLP,DNN,AE	Framework for intrusion detection in IoT netwroks Compared with ML Algorithms	UNSW-NB15 NSL-KDD99		Binary classification	Accuracy, Root Mean Square Error(RMSE),F1 Score	DNN performed well	Training and testing samples and datasets statics is not presented The research is not extended to multiclass classification Not suitable for specific types of attacks
18	[10] 2017	RNN	Intrusion detection in information security	NSL-KDD	Dos, Probe, R2L, U2R	Binary and Multiclass classification	Accuracy, DR, FPR	Achieved higher accuracy and DR and low FPR in contrast to ML-based schemes	There is still room for improvement of DR Training of model

			Compared with machine learning schemes						consumed more time Performance metrics are not graphically represented
19	[63] 2021	BiDLSMT	Intrusion detection in networks Performance comparison with other models	NSL-KDD	U2R, R2L, Dos, Probing attacks	Binary and Multiclass classification	Accuracy, Precision, Recall, F1 Score, FAR, Specificity	Achieved better results in contrast to other models	Performance improvement is required Higher Complexity Training of model consumed more time
20	[66] 2018	CAE	Dimensionality reduction based network anomaly detection Compared with other models	NSL-KDD			ROC, AUC, FPR, Detection Accuracy	Training time is reduced Less FPR Higher accuracy	Model is biased Need for the improvement of F1 Score and accuracy The research is only limited to dimensionality reduction The Model's performance is not evaluated by classifying the data according to attack types
21		SAE	Intrusion detection in network security Classified model building Parameters	UNSW-NB15	Backdoors, Fuzzers, Analysis, DoS Exploits, Generic, Reconnaissance, Shellcode, & Worms	Multiclass classification	Accuracy, DR, FPR	Performed better than existing models	Attack detection is applied only to small proportion of data Performance of the model is not checked for large proportion of data

			adjustment						Attack detection effect for small proportion attack data is lower
22	[11] 2021	HCRN NIDS	CNN capture local features RNN capture temporal features CNN – Feature extraction Classification	CSE-CIC IDS 2018	DDoS, DoS, Botnet, Brute Force , Infiltration, Web attacks, Port Scan	Binary classification	Accuracy, Precision, Recall, F1 Score. DR,FAR	Anomaly based and Signature based methods advantages are combined Computational complexity reduction Enhanced performance than other schemes	Change in signature of attached traffic leads to testing model on more datasets Testing model on latest realistic datasets is required Need to improve performance in terms of performance metrics
23	[72] 2020	MLP, CNN, LSTM, and CNN+LSTM	Cyber-attack detection in IoT Sigmoid activation function	CICIDS2017	DDoS	Binary Classification	Accuracy , Precision , Recall	Hybrid model attained higher accuracy , precision and Recall	Need for improvement in terms of accuracy and recall Training time reduction is required Performance is measured only on single attack Work is not extended to multiclass classification The number of training and testing samples and dataset statistics is not mentioned

24	[27] 2021	CNN+ GRU	Intrusion detection in IoT networks Hybridization for classification	Bot-IoT, MQTT-IoT-IDS2020, and IoT-23 intrusion detection, IoT network intrusion	DDoS, MiTM,MQTT-Brute Force, Theft, C & C, HeartBeat, OS Scan, Torii, DoS,Mirai,Sparta, FileDWNLD, Okiru, PortScan	Binary and Multiclass classification	Accuracy, Precision , Recall,F1 Score	Achieved good performance GRU resolved the issue of short term memory	Training and testing accuracy is not mentioned so the model could not be generalized Model Underfitting and overfitting
25	[62] 2020	LSTM-AE	Intrusion detection in IoT Time series features The neuron of AE is replaced by LSTM	IoT dataset published by kitsune team	ARP, Fuzzing, Mirai, SSDP Flood, Video Injection.	Binary Classification	Accuracy. Recall, F1 Score, FNR, AUC	Good performance in terms of accuracy, F1 Score, Recall , FNR and AUC	Negative samples in the dataset may affect the detection performance of the model Need to improve feature extraction method Detection of trace amount of data attacks needs to be improved Training and testing accuracies are not mentioned
26	[57] 2018	MLP-DAE	Network intrusion detection Feature selection	UNSW-NB15		Binary class classification	Accuracy, Precision, Recall, F1 Score, FPR	Low computing resource requirements Higher accuracy and F1 Score than compared schemes	Need to improve accuracy, precision, and recall Training and testing samples

			by using DAE Classification by using MLP Performance comparison with other schemes						are not mentioned Model Underfitting
27	[28] 2019	GA-DBN	Intrusion detection in IoT GA performs multiple iterations and then DBN uses optimal network structure to classify attacks	NSL-KDD	U2R, R2L, Dos, Probing attacks	Multiclass classification	Accuracy, Precision, Recall, DR,FAR	Classification accuracy improvement Reduces network complexity	Not applied in the real network environment Training and testing split percentage is not mentioned

2.4 Research Gap and Directions

Several existing Machine Learning, Deep Learning and Hybrid schemes have reviewed in the literature. As stated by the state-of-art literature we have noticed a number of drawbacks, which have not paid attention. The observed drawbacks are:

- Some of the recent existing studies did not pay attention to performance improvement of models.
- Accuracy alone is not enough to better judge about efficiency of model.
- Schemes with high biasness and low accuracy paid limited attention to training data, which results in the form of underfitting.
- Some of the existing schemes consumed much time in training.
- The hybrid schemes consumed much time in modeling.
- Inconsistency and redundancy in data leads to incorrect results or errors. To increase the accuracy and efficiency of model data preprocessing is essential.
- To enhance the performance and training sustainability of a model, data normalization is important. Data normalization has not considered in some of the schemes.
- In some of the recent schemes, no preference has given to feature selection approach for selecting the important features, which negatively affected the classification accuracy.
- Negative samples of dataset have the negative effect on the performance of models.
- Some existing schemes are limited to binary classification. Such schemes are not effective to detect known attacks.
- High False Alarm Rate (FAR) leads to reduce detection accuracy.
- Some of the schemes did not have the characteristic of generalization. It means that those schemes are not suitable for a specific type of attack. Moreover, these schemes did not have the capability to detect advanced and unseen intrusion.
- The parallel use of multiple classifier consumed increased time in modeling.
- Most of the researches are limited to performance evaluation of models. The models have not deployed in real network environment.
- Number of training and testing samples and splitting into train/test data are missing in some existing schemes. The measurement of accuracy is dependent on it.
- The old traditional intrusion detection datasets do not have the capability to handle latest attacks and real time traffic monitoring. Therefore, these datasets do not accurately measure the effectiveness of deep learning based intrusion detection in IoT.

On the basis of above gap identification, this research is giving direction towards developing hybrid ML-DL based model for detecting intrusion in IoT by using latest IoT based dataset with multiclass classification, improving the performance of models by normalizing the data and selecting the important feature. Furthermore reducing the underfitting and overfitting issues to increase the generalization of model.

2.5 Summary

This chapter has reviewed various intrusion detection schemes based on ML, DL and hybrid models. After explaining the intrusion detection in IoT-based environment, the limitations are underlined. Furthermore, this explanation undercovers that Intrusion detection is a challenge in IoT networks due to diverse nature of interconnected devices. Concise interpretation of types of ML, DL and hybrid intrusion detection in IoT is also included. After comparing different schemes major drawbacks highlighted which will improved by using the proposed scheme.

CHAPTER 3

RF-SVM-LSTM- A ML AND DL BASED HYBRID MODEL FOR DETECTING INTRUSION IN IOT

3.1 Overview

In this chapter, a novel hybrid ML and DL based mechanism has developed for detecting intrusion in IoT. The hybridization is the combination of RF, SVM and LSTM performed by using GB classifier. This hybridization has intended to detect the attacks by using latest dataset with IoT traces. The main goal of this study is to make the mechanism of intrusion detection in IoT networks effective with high accuracy. The research methodology that contains different steps described under the heading of research methodology as shown in Figure 3.1. Then the proposed scheme explained along with the steps also explained.

3.2 Research Methodology

The research methodology is comprised of four steps as shown in Figure 3.1. The first step involves selection and definition of problem, which has completed by reviewing the existing schemes for detecting intrusion in IoT based networks and gap identification in chapter 2. Furthermore, the limitations of proposed scheme presented. The second step provides the explanation of how the RF-SVM-LSTM based model has designed and developed. The third step describes the details of analysis of results in the form of tables and graphs and comparison with existing benchmark schemes. The last step is about conclusion and future work of this study.

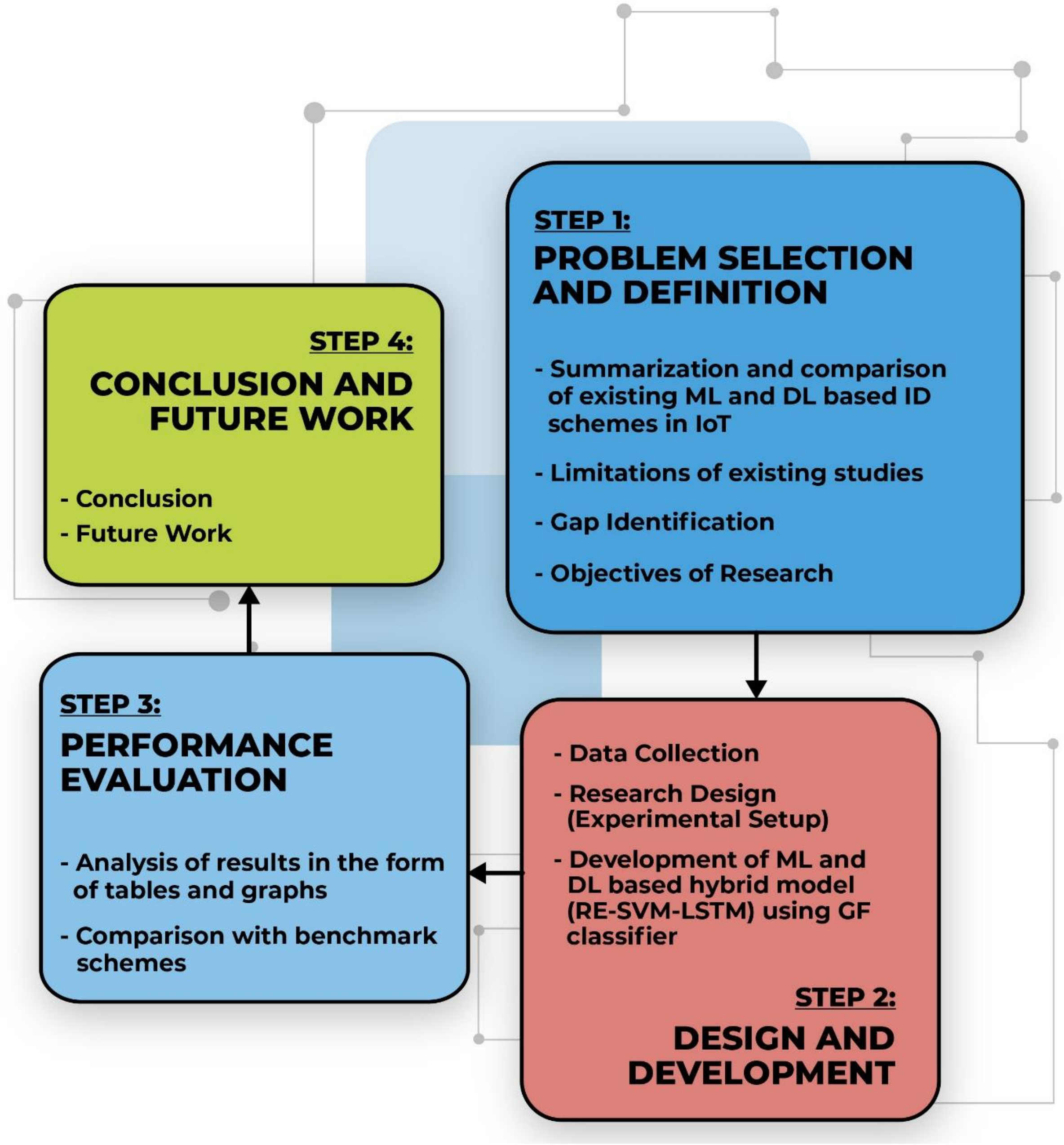

STEP 1:
PROBLEM SELECTION AND DEFINITION
- Summarization and comparison of existing ML and DL based ID schemes in IoT
- Limitations of existing studies
- Gap Identification
- Objectives of Research
- Data Collection
- Research Design (Experimental Setup)
- Development of ML and DL based hybrid model (RE-SVM-LSTM) using GF classifier
STEP 2:
DESIGN AND DEVELOPMENT
STEP 3:
PERFORMANCE EVALUATION
- Analysis of results in the form of tables and graphs
- Comparison with benchmark schemes
STEP 4:
CONCLUSION AND FUTURE WORK
- Conclusion
- Future Work

Figure 3.1: Research Methodology

3.3 Problem Selection and definition

This is the most difficult and important step of a study. Security of IoT environment has always considered a challenge because security breaching and cyber-attacks are getting worst day by day. This problem has selected and identified after presenting the Systematic Literature Review (SLR) of existing schemes used for detecting intrusion.

3.3.1 Summarization and comparison of existing schemes

The chapter 2 has summarized the information by reviewing the existing ML and DL based schemes to know what other researchers have done to identify the possible methodology for conducting this research. The comparison in the form of table draws a clear picture of selection of this problem.

3.3.2 Gap Identification

There are a number of studies proposed for the intrusion detection as discussed in the literature review section. There were a number of problems in those studies. Their drawbacks have also discussed in literature review table. The major drawbacks found out during SLR are lack of IoT related dataset availability, the performance has not evaluated by deploying in real time environment, poor performance of schemes, developing the intrusion detection mechanism for handling large-scale attacks, increasing errors of model overfitting and underfitting , lack of model generalizability, absence of data preprocessing and feature extraction from dataset. This study has proposed to fill this gap by developing the hybrid model for detecting latest attacks in IoT by using latest IoT related dataset.

3.3.3 Objectives of this scheme

The main objectives of this scheme are to develop the ML and DL based hybrid model for detecting intrusion in IoT by using IoT related latest dataset and make the mechanism of hybrid intrusion detection in IoT effective.

3.4 Research Design and Development

The research design requires the proper planning. To solve the problem of detecting intrusion in IoT the research plan answers the questions of this research by selecting the dataset and identifying the method of this research.

3.4.1 Data Collection (MQTT-IOT-IDS 2020 Dataset)

In this study, publically available IoT dataset known as MQTT-IOT-IDS 2020 dataset is used. This dataset is developed by Hindy et al. [73] This dataset is generated by using the well-known attacks and scenarios of real time devices. The MQTT network architecture has shown in Figure 3.2.

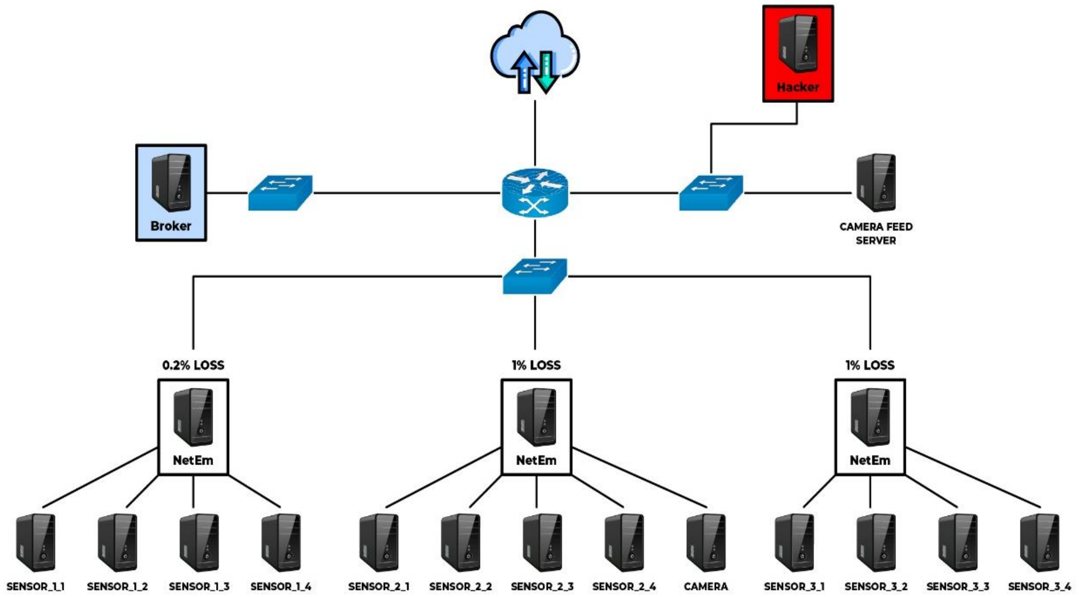


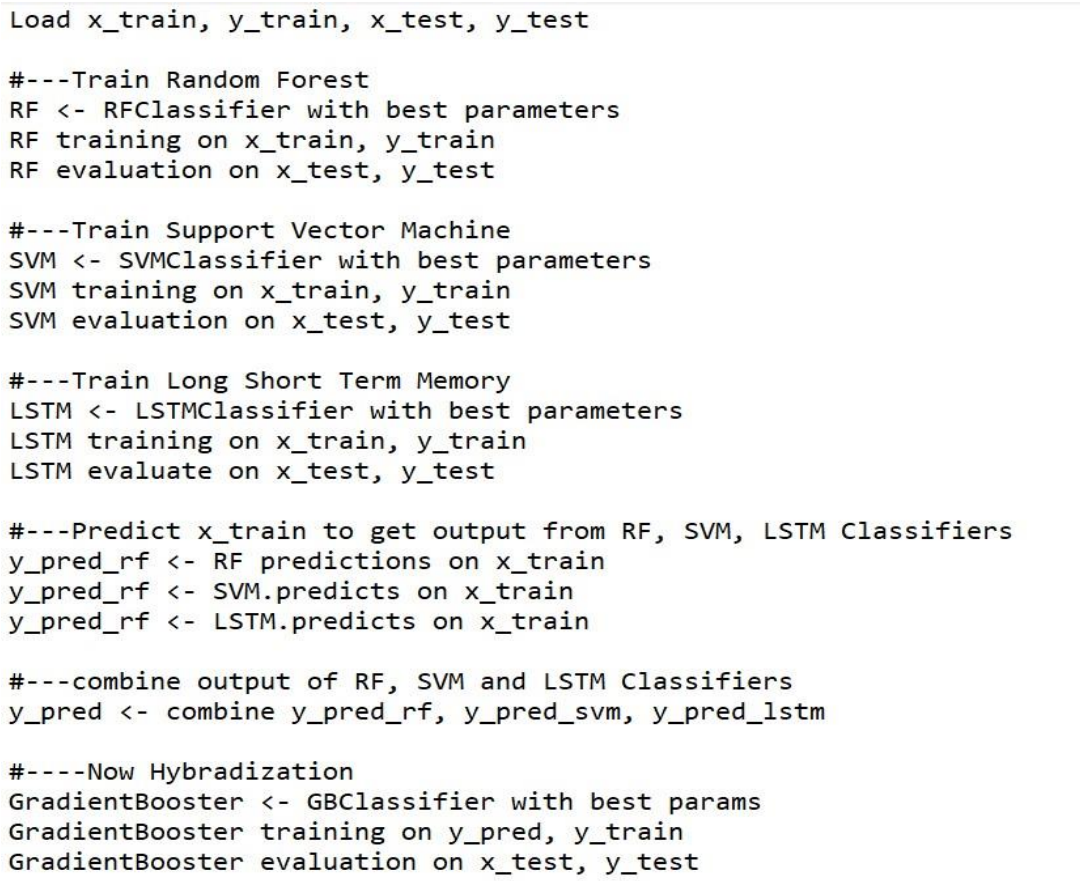
Figure 3.2: MQTT network architecture [73]

3.4.2 Research Design

Experimental design is setup by cleaning the MQTT-IOT-IDS 2020 dataset for removing the noisy data. New feature generated for improving the intrusion detection mechanism. Then label encoding and one hot encoding applied to preprocess data. Furthermore, data normalization, feature selection, feature extraction training and testing data splitting is also applied. Best hyperparameters selection based on hyperparameter tuning enhances the performance of proposed model.

3.4.3 Development of ML and DL based hybrid model

The proposed model has developed by combining three models RF, SVM and LSTM by using Gradient Booster classifier. The pseudocode of proposed model is shown in Figure 3.3.



```
Load x_train, y_train, x_test, y_test

#---Train Random Forest
RF <- RFClassifier with best parameters
RF training on x_train, y_train
RF evaluation on x_test, y_test

#---Train Support Vector Machine
SVM <- SVMClassifier with best parameters
SVM training on x_train, y_train
SVM evaluation on x_test, y_test

#---Train Long Short Term Memory
LSTM <- LSTMClassifier with best parameters
LSTM training on x_train, y_train
LSTM evaluate on x_test, y_test

#---Predict x_train to get output from RF, SVM, LSTM Classifiers
y_pred_rf <- RF predictions on x_train
y_pred_rf <- SVM.predicts on x_train
y_pred_rf <- LSTM.predicts on x_train

#---combine output of RF, SVM and LSTM Classifiers
y_pred <- combine y_pred_rf, y_pred_svm, y_pred_lstm

#----Now Hybradization
GradientBooster <- GBClassifier with best params
GradientBooster training on y_pred, y_train
GradientBooster evaluation on x_test, y_test
```

Figure 3.3: Pseudocode of proposed model

3.5 Limitations of Proposed scheme

The proposed model is evaluated on a single dataset. Number of attacks handled in the proposed scheme are limited. More attacks could be handled in real healthcare environment for future research.

3.6 Proposed scheme (RF-SVM-LSTM- A ML and DL based hybrid model for detecting intrusion in IoT)

The proposed study have used hybrid ML/DL based ID in IoT devices at healthcare environment. Figure 3.4 explains the working methodology of the proposed model.

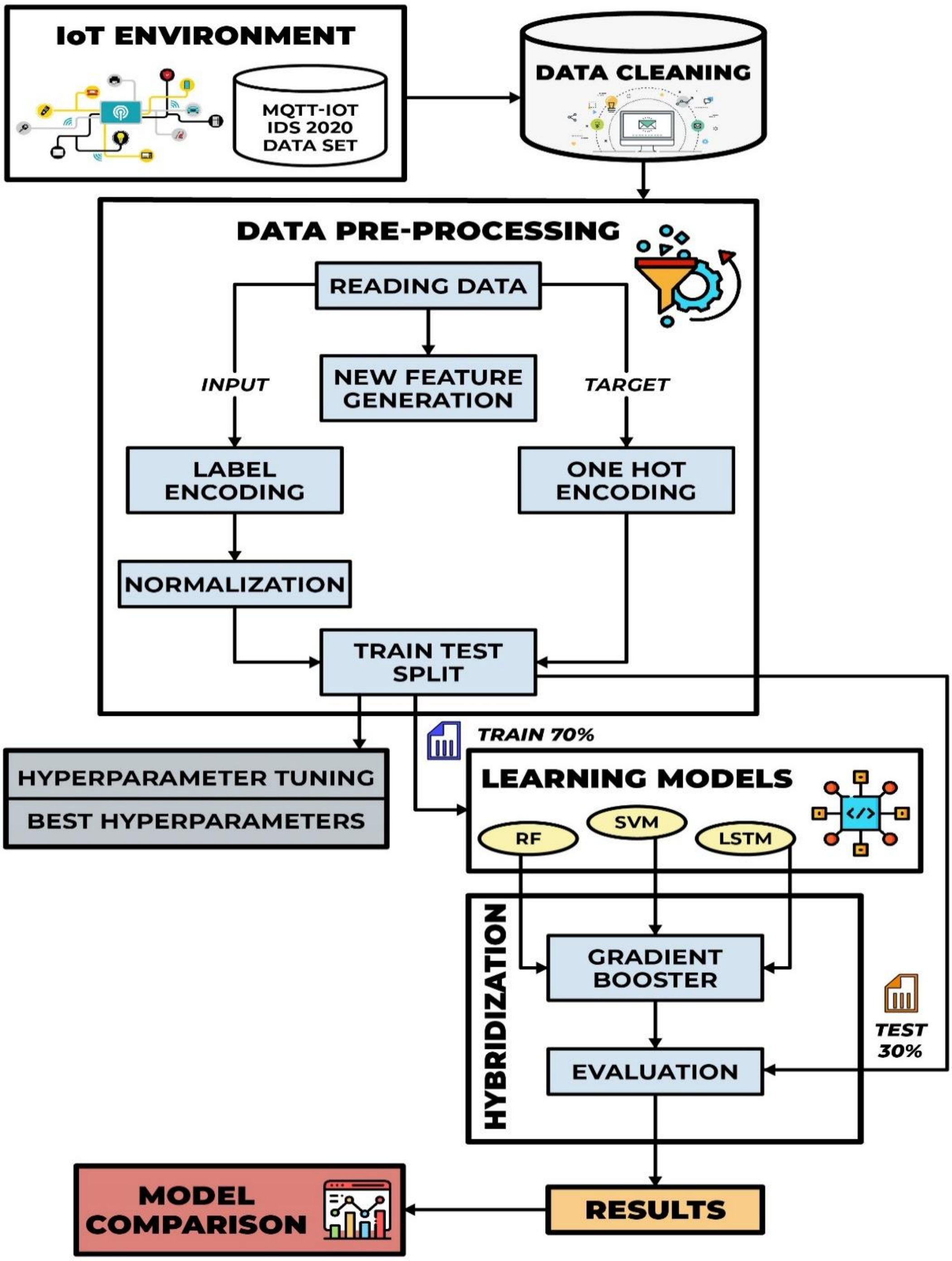


Figure 3.4*:* Experimental Setup

The systematic working discussed below:

1. In the first step MQTT-IOT-IDS 2020 dataset is downloaded from IEEE website MQTT-IOT-IDS 2020 [74].
2. In the next step, the data is preprocessed and main useful features from the data extracted.
3. Data has divided into two groups training and testing.
4. Hyperparameter tuning has applied for selecting best hyperparameters.
5. Evaluation matrix used for performance calculation. The evaluation matrix used for the study includes accuracy, precision, recall and F1 score.
6. The outcomes of the proposed model is compared with the other models.

3.7 Data Preprocessing

Data preprocessing is the process of taking the data from the dataset and transform the data into the format that is readable by the machine. It is the most important step of machine learning process. For the proposed study, data preprocessing helps the machine-learning algorithm for the improvement of their accuracies and results. The process of data preprocessing carried out in different steps. These steps are

1. Data Cleaning
2. Data transformation (Feature extraction, Normalization)
3. Data reduction

3.7.1 Data Cleaning

Data cleaning is the process of data repairing for the purpose of experimental analysis. Data quality assessment is included in this step. It describes the selection of a high quality accurate dataset for training and testing the machine learning algorithms. A good dataset impact on the overall quality of the project.

This process includes the conversion of raw data formats into readable machine and human codes. The repairing is based on modifying or removing the data that is wrongly formatted [75]. Before jumping into the processing of the data, all the machine-learning algorithms need the process of data cleaning for further use. For the proposed study the dataset is taken from MQTT Internet of things data intrusion website [74]. "preprocessing. Labelencoder ()" deep learning function is used in this study for data cleaning purpose. This is the latest dataset developed for intrusion detection.

The focus of this study was to use a high quality dataset for intrusion detection for IoT healthcare device. Dataset is the baseline of any study. The dataset consists of five classes includes four type of IoT attacks and one non-attack scenario.

1. Normal operation
2. Sparta SSH brute-force
3. MQTT brute-force attack.
4. Aggressive scan
5. UDP scan

The dataset contains the pcap file for five classes named as normal.pcap, sparta.pcap, scan_A.pcap, mqtt_bruteforce.pcap and scan_sU.pcap. These file contain thousands of data sequences regarding IoT attach. The dataset present on the MQTT website is in categorical format. For training the dataset into the machine-learning algorithm, the data converted into numeric values. The reason is all the machine learning algorithm process the numeric data easily. Machine learning algorithms used csv files for data processing. So there need to be convert the data into csv for further using scenario.

For this purpose, we first download the csv data file from the MQTT dataset website. Machine learning libraries Matplotlib, Pandas and Numpy are used in this model for data cleaning purposed. The cleaning of dataset include Data Normalization, Removing Unicode, Removing strop words, Stemming, Lemmatization, POS, and Sorting. In the proposed study the dataset was in tabular form, so the cleaning process includes

- Identification of the rows and columns that has very less number of data.

- Identification of the rows and columns that has repeated values.
- Identification of the rows and columns that has single values.
- Identification of the rows and columns that has low variance.

After the identification of the rows and columns mentioned above, there need to balance or remove that data from the rows and columns for further processing. The missing data from the dataset is sorted by ignoring the tuples or by adding the values manually. Synthetic monitoring oversampling Synthetic monitoring resampling and Synthetic monitoring under sampling techniques are the most commonly used machine learning techniques for balancing the data. In the under sampling strategy of data balancing the instances of dataset are reduced. In oversampling strategies, the instances of minority classes are increased and in resampling methods, the instances are resamples according to the available data instances. For this study, we are using synthetic monitoring oversampling method for data cleaning and balancing.

In this study, we are using oversampling techniques for data balancing. SMOTE is one of the most commonly used data balancing strategy for noisy and unbalanced data in machine learning. SMOTE creates the synthetic data points in data augmentation [76]. SMOTE algorithms for data balancing is stated as follow

1) Create the majority and minority classes of the dataset.
2) Mark oversampling instances.
3) Find out the i instance from the minority class.
4) Identify the nearest neighbor n of the instance i.
5) Find the distance between n and i.
6) Find the product of the distance d with any digit between 0 and 1.
7) Repeat the steps until we find the required instances.

3.7.2 Feature extraction

After balancing the dataset by removing the noise and missing values, the dataset is process for feature extraction. Feature extraction is the process of selecting the most valuable feature of your data for the further processing. The selected variables of the features are further process to ML algorithms that reduce the processing time for the algorithms. . In Machine learning algorithms pattern recognition is one of the most important concept of feature extraction because from these patterns the algorithms perform training and testing process [77]. We are using two method of feature extraction in the proposed study

- Label encoding method
- One hot encoding method

Label encoding method is the feature extraction method used in the proposed study [78]. This method is also handle with machine learning library Sklearn. Label encoding method assign each categorical value of the dataset into integers based on the alphabetical orders. The label encoding method is used in the proposed study for the conversion of each data labels into different numeric values. All the labels having the values starting from 0. The problem of using label encoding for the proposed study is the generation of priority issue. The label with low priority may be neglected while processing. This is the reason that we are using one hot encoding method with label encoding.

One hot encoding is the feature extraction method that converts the feature vectors into the encoding vectors. This feature extraction process converts the categorical data into the feature vector format that is readable by the machine-learning algorithm for increasing the prediction accuracy. [79]. This method add new column in the categorical data with binary values. The values are represented by 0 and 1. Where 0 represents false and 1 represents true. One hot encoding is very effective in intrusion detection scenario because the machine learning algorithm LSTM, RF and SVM treat the order of the number as the attribute of the significance. As not all the values in MQTT dataset are ranked so there will be problem for the ML algorithms in poor performance and prediction. One hot encoding method rank the values in the MQTT dataset and improve the efficiency of performance in the dataset. We are using machine learning library Pandas and Sklearn for one hot encoding method. The reason behind using these libraries in one hot encoding method is that these are the open source libraries used for both deep learning and machine learning algorithms and provide versatile and powerful. These are the machine learning libraries that provide the tools for selection, clustering, classification and regression problems.

One hot encoding is also used in machine learning problems for overcome the gradient vanishing problem. As it is very difficult for the machine learning algorithms to treat with very high and very low values. When the values are mapped with 0 and 1, the gradient vanishing problem is tackled hence the accuracy of the model improves. The process of One hot encoding consists of three steps. In the first step, we convert MQTT categorical dataset is imported using pandas. In the next step the numeric values are assigned with the binary values using one hot encode. In the last step, we drop the original categorical values.

The equations for one hot encode vectors is represented by equation (3.1).

$$\mathbf{o} \in \{0, 1\} \quad \sum_{i=1}^{m} o\mathbf{1} = \mathbf{1} \qquad (3.1)$$

In the equation o is vector one hot encoder, m is the length of the vector. There are five classes of data in the proposed study as normal, sparta, scan_A, mqtt_bruteforce and scan_sU. One hot encoding method converts these data classes into the encoder vector. Figure 3.5 explains the feature selection after the feature extraction models in the proposed scenario

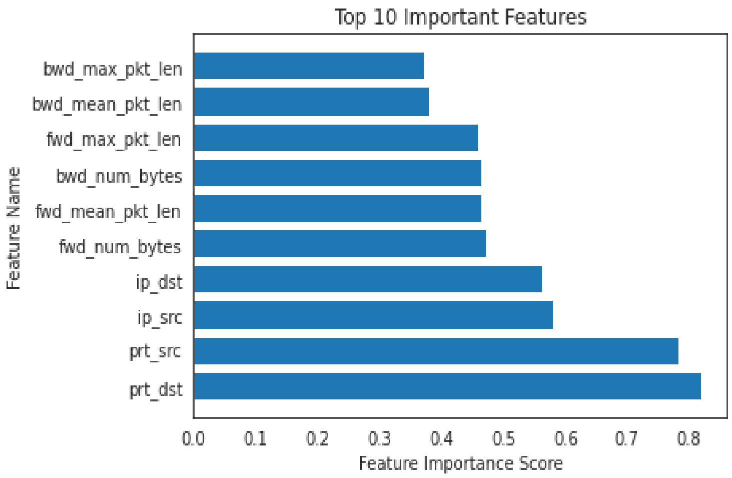


Figure 3.5: Top 10 Important Features

After the completion of feature extraction method the data is normalized for using it to the further processing. Data Normalization is technique of data preparation for using it for the machine learning algorithms. This process change the numeric data (data after feature extraction method) to common scale without losing any information. The process of data normalization include

- Add the data to be normalized and find the machine learning components for data transformation.
- Choose the columns to be normalized.
- Make sure not to choose the columns with single unchanged numeric values.
- Choose mathematical function Zscore, MinMax, Logistics, LogNormol, and tanh for the normalization process.
- Visualize the data after normalization.

The equations used for the process of data normalization are

$$\boldsymbol{z = \frac{x - \min(x)}{stdev(x)}} \qquad \textbf{(3.2)}$$

$$\boldsymbol{z = \frac{x - \min(x)}{max(x) - min(x)}} \qquad \textbf{(3.3)}$$

$$\boldsymbol{z = \frac{1}{1 + exp(-x)}} \qquad \textbf{(3.4)}$$

In the equation x is the selected data instance in the column. After the process of data preprocessing the dataset is categorized in training and testing dataset for tackling the model over fitting and under fitting problem. The proposed study is using 70% dataset for the training purpose and 30% data for testing purpose.

3.8 Prediction Algorithm

For the proposed study we are using machine learning and deep learning algorithms such as Random forest, Support Vector Machine and Long term short term memory network for model training and testing purposed for intrusion detection. The explained prediction algorithms used in the current process is defined in the section below.

3.8.1 RF

Random Forest (RF) is one of the most commonly used supervised ML algorithm for prediction, regression and classification problems. This is a tree based prediction algorithm handle the missing data for the model and avoid the model over fitting [80]. The higher the number of trees in the dataset the lower probability of the model over fitting and higher the accuracy. One of the main reason for the selection of random forest algorithm for intrusion detection system is that the algorithm is efficient in providing feature selection, proximity metrics and classification. The algorithm dhows the low training complexity *(O (n (log (n))*) in terms of intrusion detection, resilience in term of dealing with the imbalanced data, able to deal natively with categorical and continuous features.

The prediction model of random forest algorithm is based on the decision tree algorithm results. The resampling model for intrusion detection is inspired by the bootstrap approach that is used for the creation of the tree-structured forest. The RF model tuned on two structures [81]

mtry: Randomly selected features selected for each split.

ntree: The total number of trees in the RF model.

In the proposed scenario, RF takes the multiple inputs from the dataset and combine the multiple outputs of the classification to gives the output. RF combines the multiple classifiers to solve the complex problems. RF contains a many decision trees on several subgroups of the MQTT dataset and takes average to improve the predictive accuracy of that dataset. RF takes less number of inputs in the tree as compare to the other machine learning algorithms. The predicted output accuracies for the RF are high as compared to the other algorithms even for the larger problems like ours. As in our scenario, we have thousands of data records in which many data records are missing, RF can maintain the high accuracies in that scenario too. The algorithm for RF for the proposed dataset of MQTT is

1) Select of the random instance from the training dataset of MQTT dataset and named as k.
2) Make the subset of the k by considering the nearest data points and create a tree.
3) Make N number of tree according to the dataset.
4) Repeat step 1 and 2 for all data trees.
5) For every data instance k in dataset, find the result of each decision tree, and give the new data points to the category that have the majority classes, as it is an attack on non-attack scenario and name them.

For the proposed study, Randomforest classifier() and sklearn.ensemble() libraries are used for working with random forest algorithm. The python code used for Random forest classifier is

```
from sklearn.model_selection import RandomizedSearchCV
from sklearn.ensemble import RandomForestClassifier
params = {'bootstrap': [True, False],
'max_depth': [10, 20, 30, 40, None],
'max_features': ['auto', 'sqrt'],
'min_samples_leaf': [2, 4, 6],
'min_samples_split': [2, 5, 10],
'n_estimators': [20, 40, 60, 80, 100]}
rf = RandomForestClassifier()
rf_random = RandomizedSearchCV(estimator = rf, param_distributions = params, n_iter = 1, cv = 2, verbose=2, random_state=42, n_jobs = rf_random.fit( x_train, y_train )
```

The **‘bootstrap’** is used to train each data point at least once in the RF algorithm. After using the bootstrap parameter, every data instance used by the random forest at least once. The data instance may be used more than once in a tree node. The maximum depth of nodes in the structure of tree is demonstrated by **'max_depth'.** For the proposed study, we use the maximum depth of 10 to 40 for each node of the tree. **'max_features'** is used to determine the maximum number of features that are utilized by a single tree in RF that is set to be auto. The minimum number of samples required to split in the node are measured through the '**min_samples_split'** parameter. Moreover, the minimum number of samples are estimated through '**min_samples_leaf** 'parameter. We set these values from 2 to 6. It implies that the model have atleast 2 to 6 parameters at the leaf node according to the output. The **'n_estimators'** use in the code is used to indicate required number of trees in the proposed algorithm. The estimated trees are about 20 to 100. From the proposed model, the results of accuracy, precision, recall and F1 score is calculate.

MSE measure the average of the square of errors in the RF algorithm. MSE is the between the calculated values and actual values. The mean square error in RF is measured by

$$\mathbf{MSE} = \frac{1}{N}\sum_{I=1}^{N}(f1 - y1)^2 \qquad \textbf{(3.5)}$$

In the equation $(f1 - y1)^2$ is the square of errors. Where y1 is the predicted values and f1 are the actual values. Entropy is the measure the uncertainty and disorder in the results represented in equation 1.6.

$$\textbf{Entropy} = \sum_{I=1}^{C} -p1 * log_2\, \mathrm{p1} \qquad \textbf{(3.6)}$$

In the equation p1 is the prior probability of each class, c is the number of unique classes [82]. Figure 3.5 explains the working of Random forest algorithm for intrusion detection in health care IoT devices.

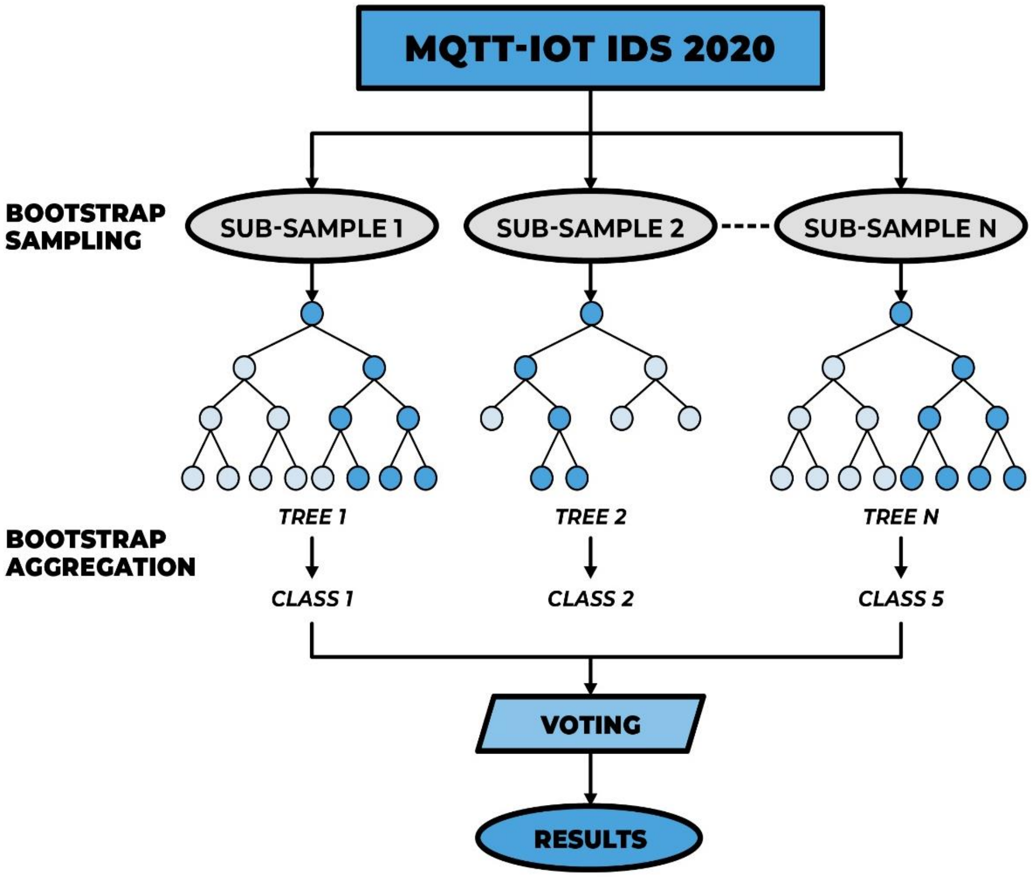


Figure 3.6: Random Forest algorithms for MQTT-IOT-IDS 2020 dataset for intrusion detection model

As the figure 3.6 explains, RF algorithm selects the data from the dataset. Different parent classes are defined for the randomly selected dataset. Their child nodes are created. We have about 20 to 100 parent nodes in the EF algorithm. After processing different features and parameters from the child nodes the data pass to the leaf node classes where the result are categorized with respect to the data classes. As mentioned earlier in the section we have five output classes, One for the non-attack and four for the different types of attack. The RF algorithm classify these dataset according the valued output.

3.8.2 SVM

Support vector machine is also a supervised ML algorithm widely used for classification and regression problems. The aim of SVM classifier is to determine the hyperplane that classify the data points based on their features [83]. A best hyperplane is the one that efficiently distribute the classes. For the proposed study, we are using four hyperplane for the distribution of five different classes of IoT errors. For the proposed study, we are using SVM due to its efficiency in high dimension cases. It use the memory in an efficient manner and its kernel trick.

SVM kernel is the biggest strength of SVM algorithm. It takes the low dimension data, converts it into the high dimension, and converts the non-spreadable data into separate one. The step by step working of the SVM for intrusion detection is discuss below

1. Load the different libraries for SVM classifier in Python. In the proposed study we are using Sklearn and pandas for SVM algorithm

   ```
   import pandas as pd
   import Sklearn
   from sklearn.svm import SVC
   from sklearn import metrics
   ```

2. In the next step the dataset is imported and features of the dataset are extracted.
3. The dataset is converted into training and testing.
4. Initialization of SVM kernel. The following code is used for SVM kernel initialization in the proposed study

   ```
   params = { 'C' : [1, 2, 3],
   'kernel' : ['linear', 'poly', 'rbf', 'sigmoid'],
    'degree' : [5, 10, 30],
   'gamma' : ['scale', 'auto'],
    'decision_function_shape' : ['ovo', 'ovr'] }
   ```

Parameter C in the python code is used to avoid the over fitting of the SVM model. These regularization parameters are used for overcoming errors. There are different types of kernel we are using for SVM classifier as Linear, Poly, Gaussian and Sigmoid. These kernels are used to map highly dimension features. These kernels are used when there is no prior knowledge of the data or used as the proxy of NN. The equations of these kernels are

$$\boldsymbol{Linear\ kernal}:\ k(x,y) = 1 + xy + xy\,min(x,y) - \frac{x+y}{2}\,min(x,y)^2 + \frac{1}{3}\,min(x,y)^3 \quad \textbf{(3.7)}$$

$$\boldsymbol{Polynomial\ kernal}:\quad k(x_i, x_j) = (x_i +\ x_j + 1)^d \quad \textbf{(3.8)}$$

$$\boldsymbol{RBF\ kernal}:\quad k(x,y) = exp\left(-\frac{||x-y||}{\alpha}\right) \quad (\mathbf{3.9})$$

$$\boldsymbol{Sigmoid\ kernal}:\quad k(x,y) = \tanh(\alpha x^T y + c) \quad (\mathbf{3.10})$$

In the equations x, y are the data points at the hyperplane k**.** The degree parameters are assigned for the polynomial kernel. This value is ignored by the other SVM kernels. The gamma value is used for poly, sigmoid and rbf kernels. One vs one (ovo) and one vs rest classifier (ovr) decision functions are use in the SVM code for explaining the predictions of the points in the hyperplane.

5. Fitting SVM classifier.

```
svc_random = RandomizedSearchCV(estimator = svc,

param_distributions = params, n_iter = 1, cv = 2, verbose=2,

random_state=42, n_jobs

svc_random.fit( x_train, np.argmax( y_train, axis=1 ) )
```

6. Generation of the results
7. Evaluation of performance of the model.

```
y_pred = svc_random.predict(x_test)

y_true = reverseCoding( y_test)

print ( "Accuracy:", accuracy_score(y_pred, y_true))

print( "Recall:", recall_score(y_pred, y_true, average="weighted") )

print ("F1 Score:", f1_score(y_pred, y_true, average="weighted") )

print( "Precision:", precision_score(y_pred, y_true, average="weighted") )
```

The results are generated by using the evaluation matrix Precision, Accuracy, Recall and F1 Score. The output of the results are discussed in Results and Discussion chapter. Figure 3.7 explains the working of SVM model for the proposed study

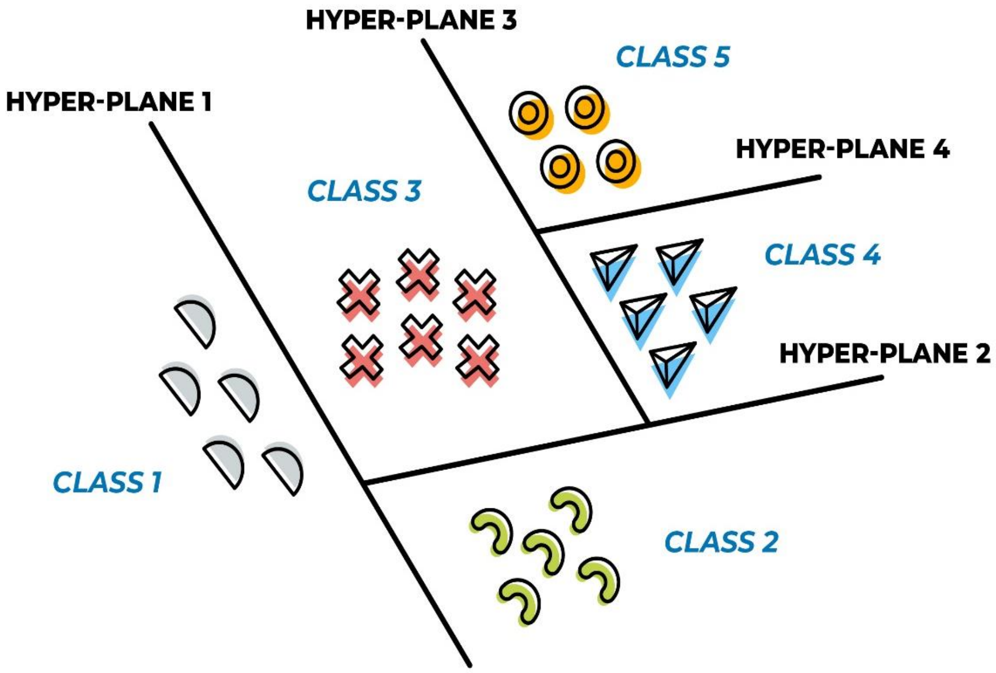


Figure 3.7: SVM algorithms for MQTT-IOT-IDS 2020 dataset for intrusion detection model

Figure 3 shows the five classes of the output with four hyperplane. The hyperplane parameter distinguished the result classes from each other. These five SVM classes are categorized into attack and non-attack scenarios for IoT devices.

3.8.3 LSTM

Long Short Term Memory (LSTM) is the third algorithm used in this study for intrusion detection. It is a deep learning algorithm inspired by the gated cell. It is consists of input layer, hidden layers and output layer. In every LSTM cell, there are three gates, Input gate for accepting the input from the previous LSTM cell, Forget gate that process the data and decide which data should keep and which should discard. The output layer that pass the results to the next LSTM cell [84]. These gates are useful for overcoming the vanishing gradient problem. There are wide ranges of parameters used inside the LSTM cell includes input and output biases, learning rates and evaluation functions that make LSTM performance better than other algorithms. LSTM cells are connected with each other and the gates helps the LSTM architecture to regulate the needed information. A structure of LSTM cell used in the proposed study is explained in Figure 3.8

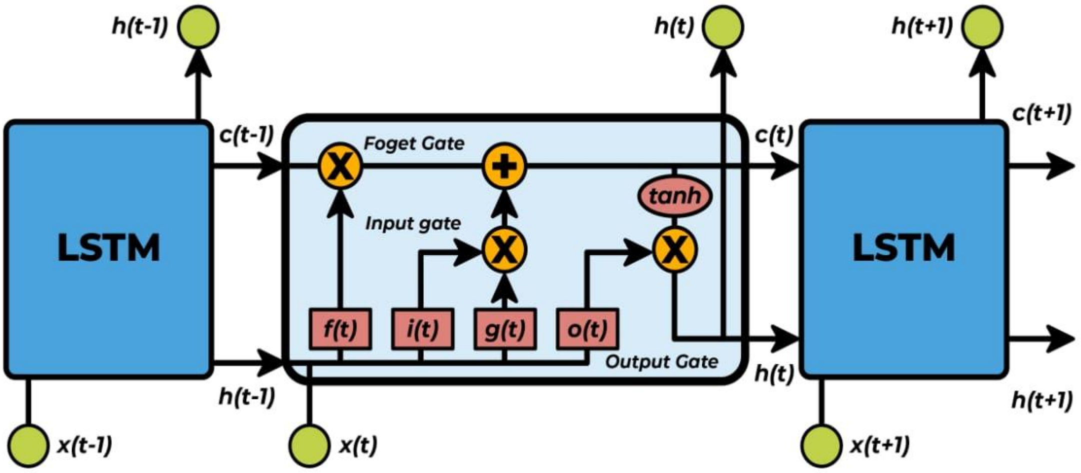


Figure 3.8: Structure of LSTM cell

From the figure is seen that x is the input of LSTM cell at specific time t represented by x_t , and h is output at specific time t denoted by h_t. f_t is the forget gate, i_t represent input gate and o_t shows the output gate. h_{t-1} .Moreover, C_{t-1} are the outputs from the previous LSTM cells that are used in the next cell as input. The equations of LSTM cell are

$$i_t = \sigma\left(y_t U^i + h_{t-1} W^i\right) \qquad (\mathbf{3.11})$$

$$f_t = \sigma\,(y_t U^f + h_{t-1} W^f) \qquad (\mathbf{3.12})$$

$$o_t = \sigma\,(x_t U^o + h_t W^o) \qquad (\mathbf{3.13})$$

$$C_t{}' = tanh\,(x_t U^c + h_{t-1} W^c) \qquad (\mathbf{3.14})$$

$$C_t = \sigma\,(f_t * C_{t-1} + i_t * C_t{}' \,) \qquad (\mathbf{3.15})$$

$$h_t = \tanh\,(C_t) * o_t \qquad (\mathbf{3.16})$$

The gates in LSTM cell works as sigmoid function and tanh function is applied in the last layer of LSTM. Sigmoid is used for plotting the values between 0 and 1 while tanh activation is used to plot the values between -1 and 1. Many mathematical functions are applied inside these activation functions for maintaining the values between the desire axes for regulating the output for the cell.

Both of these activation functions in LSTM cell is responsible for forget and pass the data. The value of 0 in the sigmoid represent that the data is not too important for the next iteration and to be forgetter while 1 represents the valuable data that is needed to be processed forward. In the proposed study keras tuner library is used for picking the optimal hyper parameter sets for the LSTM cell using tensor flow. The working of LSTM cell for the proposed model of detection intrusion is explained as follow

- Takes the input from the previous LSTM cell in intrusion detection dataset.
- Calculate the values of each gate using different weights and apply different activation functions.
- Calculate current cell state. This is calculated by
    - Calculating the element-wise multiplication vector of the input gate and the input modulation gate
    - Calculating the element wise multiplication vector of the forget gate and the previous internal cell state and then adding the two vectors.
- Compute the hidden cell state by forcing the element wise hyperbolic tangent of current internal cell state and then multiply it with output gate.

The LSTM python code used for the proposed study is

```
def model_iot(hp): hp_units = hp.Int('units', min_value=32, max_value=512, step=32)
act = 'relu'
model = Sequential([
layers.Dense(hp_units, activation='relu'),
layer layers.LSTM(100, activation='relu', return_sequences=True ),
layers.Dense( 5, activation='softmax' )] )
hp_learning_rate = hp.Choice('learning_rate', values=[1e-2, 1e-3, 1e-4])
optm = Adam(learning_rate=hp_learning_rate)
model.compile(loss='categorical_crossentropy',
```

x. optimizer=optm, metrics=['accuracy'])

xi. return model

Sequential model of deep learning is used for LSTM algorithm alongwith 32 minimum data input and 512 maximum data values for one iteration of working. LSTM also work as back propagation algorithm. We are using one input layer, two dense layers, one LSTM layer and one output layer in LSTM algorithm. In the first dense layer Relu activation function is applied. Relu works in the principal of min and max. If the value of the function is negative it will be considered as zero and the function with positive values are considered as one. In the other dense layer, softmax activation is applied that determine the probability of the class from where it belongs and send it to the output layer. The formula for softmax probability calculation is explained in equation (3.17)

$$\alpha(z)_j = \frac{e^{\mathrm{z}^{(j)}}}{\sum_{k=1}^{k} e^{\mathrm{z}^{(k)}}} \text{ for j=1 to k} \qquad (\mathbf{3.17})$$

In the equation$(z)_j$ is the input vector, α represents softmax function, $e^{\mathrm{z}^{(j)}}$ is the exponential function for input and $e^{\mathrm{z}^{(k)}}$is the exponential function for output, where k is the number of classes. Learning rate are applied after the dense layer to increase the accuracy and for minimum loss. Adam optimizer is used at the last of the LSTM cell for reducing the loss from each iteration and improving the model accuracy. Adam is the mostly used optimizer because it gives best results, takes fewer parameters for tuning and have faster computational time. Figure 3.8 explains the LSTM layer for the Intrusion detection.

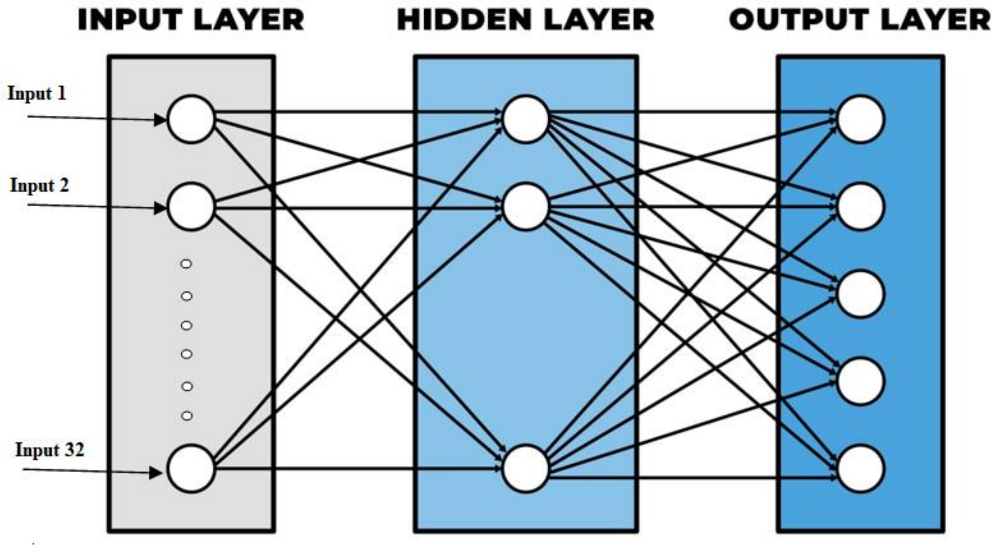


Figure 3.8: LSTM Architecture

As shown in the Figure 3.8, we have 32 input neurons in each iteration of LSTM model and 5 output layers at the last.

5.9 Summary

In this chapter of our research, we discuss the working methodology for the Intrusion detection in IoT based healthcare. We discussed about the two ML and one DL model as Random forest, Support vector machine and LSTM used in the detection and classification problem for the proposed study. We deeply discussed all the algorithms, their working in the proposed study along with the implemented python code. The results of these algorithms along with their graphs and accuracies are discussed in the next chapter.

CHAPTER 4

PERFORMANCE EVALUATION

4.1 Overview

In this chapter, we discuss the performance evaluation of the proposed hybrid ML and DL based model for detecting intrusion in IoT based smart healthcare. This chapter consists of two sections. In results section, performance of proposed model is evaluated using difference performance metrics. In analysis section, comparative analysis of proposed model with existing schemes has presented.

4.2 Results and Analysis

This chapter presents the implementation of proposed model. Intrusion detection mechanism is evaluated through various performance metrics. We presented the results of hybrid RF-SVM-LSTM model's performance through different metrics. The performance of proposed model is improved by tuning hyperparameters and selecting best hyperparameters. The metrics used to evaluate the performance discussed below:

4.2.1 Accuracy

The ratio of accurately classified records to overall number of records is defined as accuracy. In case of high accuracy, ML and DL models considered as better performing models.

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

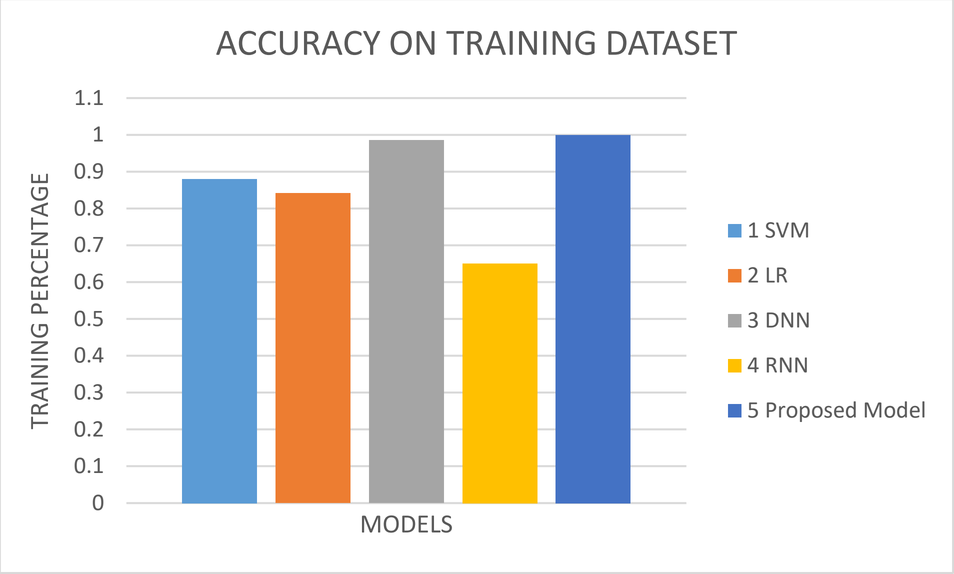


Figure 4.1: Accuracy of models on training dataset

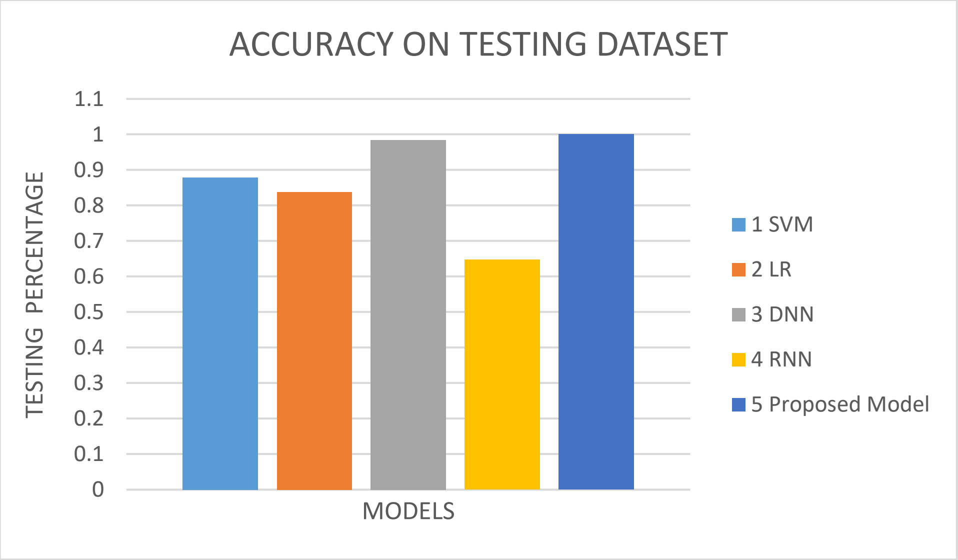


Figure 4.2: Accuracy of models on testing dataset

4.2.2 Precision

Precision is defined as the ratio of accurate detection of records that contains attacks to the entire range of all detected attacked records.

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

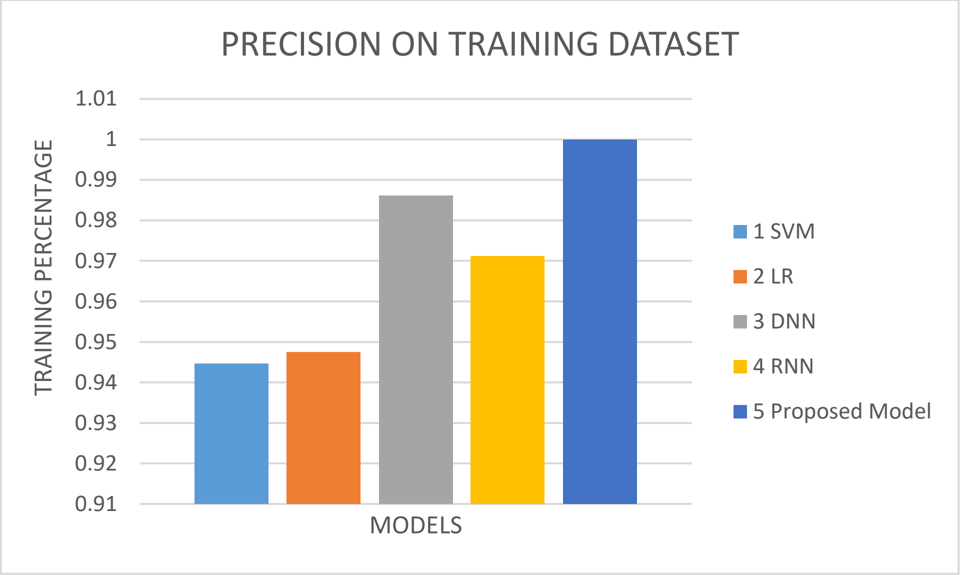


Figure 4.3: Precision of models on training dataset

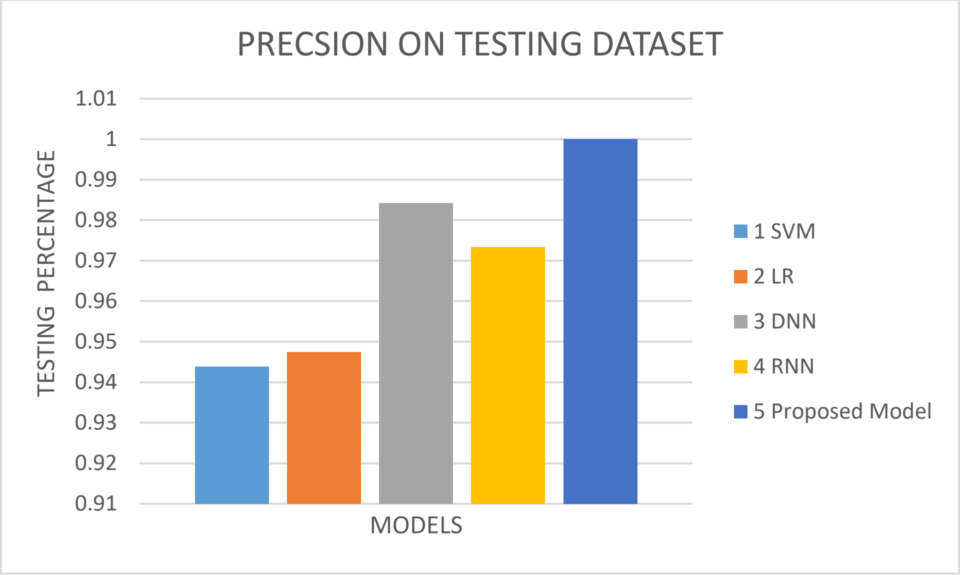


Figure 4.4: Precision of models on testing dataset

4.2.3 Recall

Recall is defined as the ratio between true positive and sum of true positive and false negative.

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

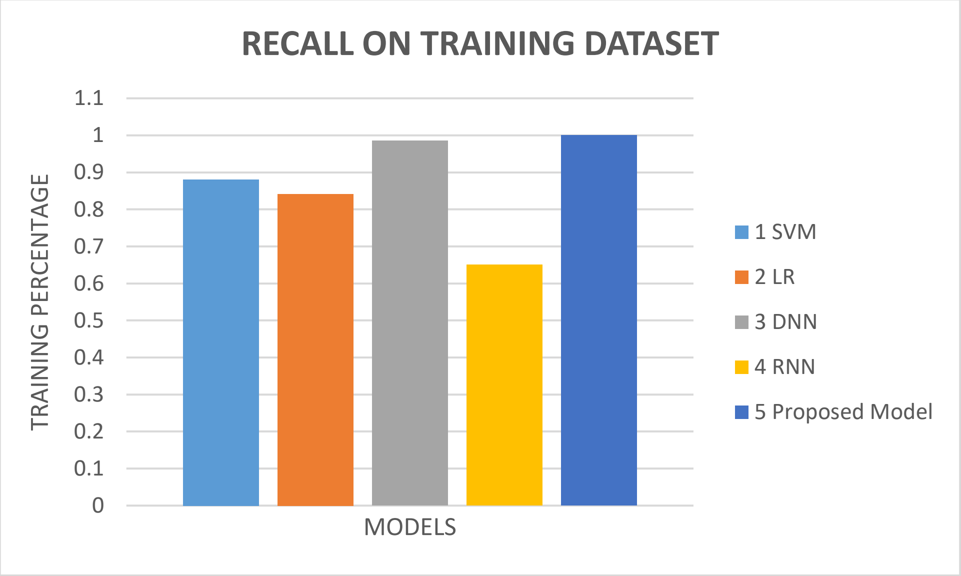


Figure 4.5: Recall of models on training dataset

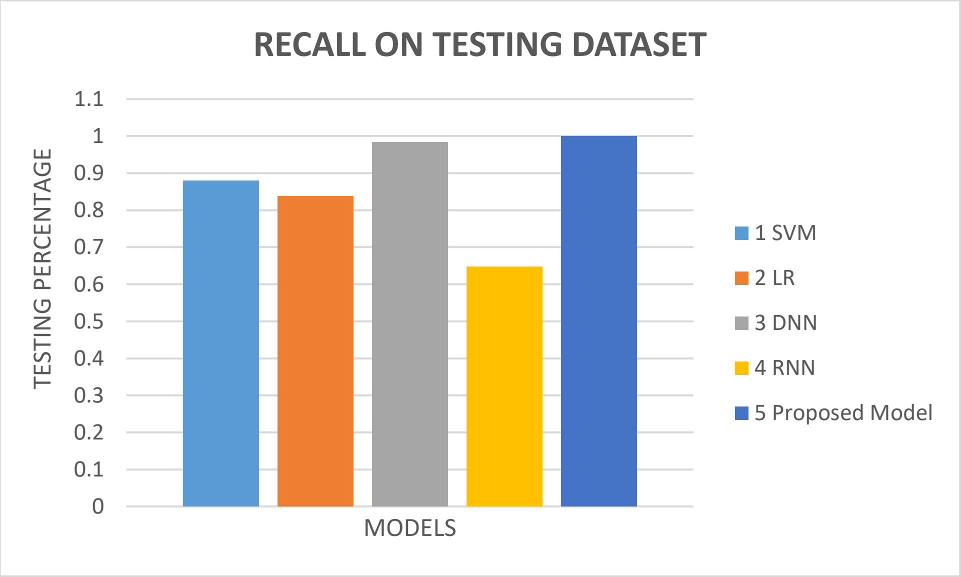


Figure 4.6: Recall of models on testing dataset

4.2.4 F1-Score

F1-Score is the weighted average of recall and precision.

$$F1 - Score = \frac{2 * \ TP}{2 * TP + FP + FN}$$

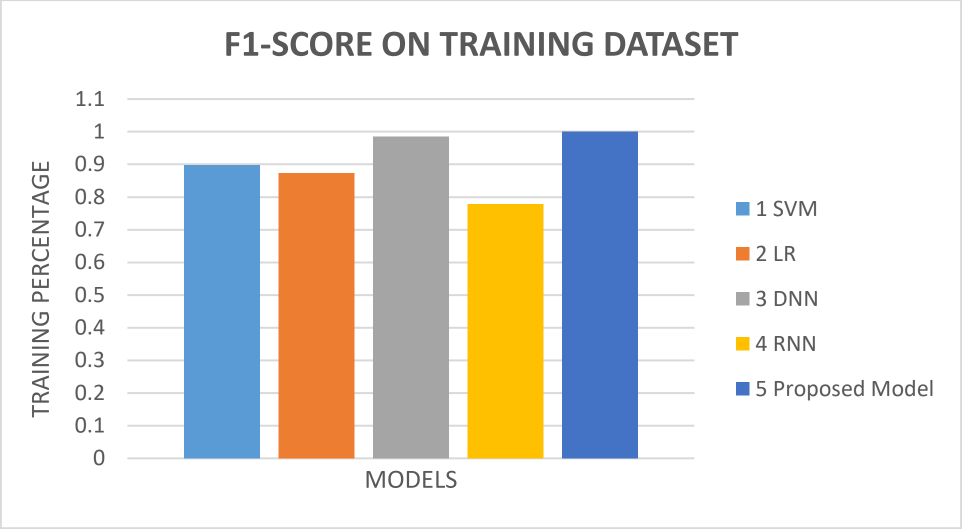


Figure 4.7: F1-Score of models on training dataset

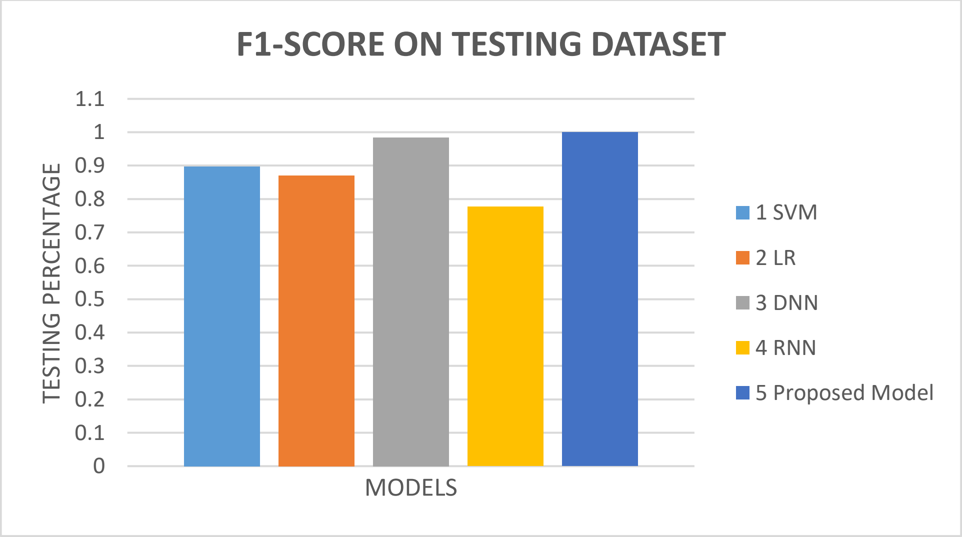


Figure 4.8: F1-Score of models on training dataset

4.2.5 Performance evaluation of models on training dataset

Table 4.1: Performance Evaluation of models on training dataset

SR. #	MODEL	TRAINING ACCURACY	TRAINING PRECISION	TRAINING RECALL	TRAINING F1-SCORE
1	SVM	0.8804	0.9446	0.8804	0.8984
2	LR	0.8419	0.9474	0.8419	0.8735
3	DNN	0.9855	0.9861	0.9855	0.9856
4	RNN	0.6507	0.9712	0.6507	0.7787
5	Proposed Model	1.00	1.00	1.00	1.00

4.2.6 Performance evaluation of models on training dataset

Table 4.2: Performance Evaluation of models on training dataset

SR. #	MODEL	TESTING ACCURACY	TESTING PRECISION	TESTING RECALL	TESTING F1-SCORE
1	SVM	0.8792	0.9439	0.8792	0.8975
2	LR	0.8381	0.9475	0.8381	0.871
3	DNN	0.9839	0.9843	0.9839	0.9839
4	RNN	0.6474	0.9734	0.6474	0.7769
5	Proposed Model	1.00	1.00	1.00	1.00

4.2.7 Correlogram

Correlogram describes the relationship between features. As shown in Figure 4.9, most of the features have highly positively correlated with each other. It means that these features have highly positively correlated with target variables.

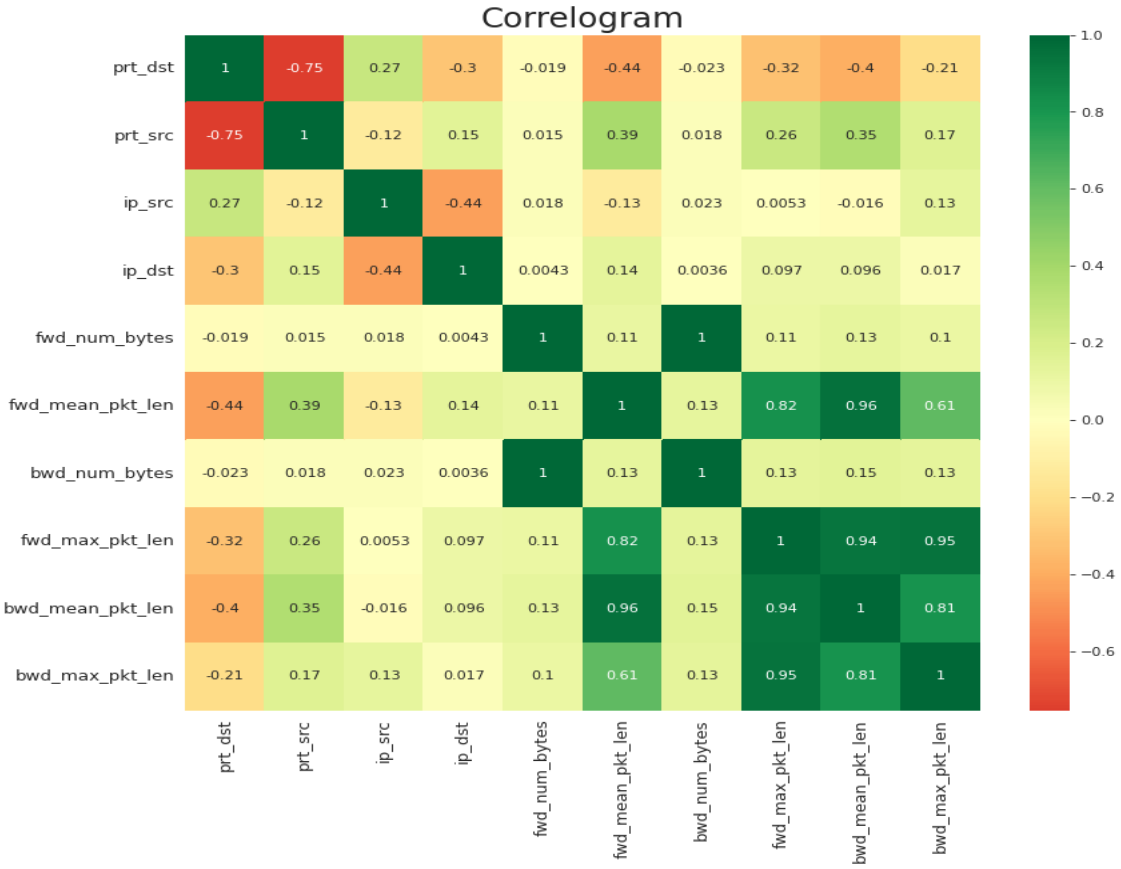


Figure 4.9: Correlogram

4.2.8 Confusion matrix

Confusion matrix defines and summarizes the performance of classification algorithm. A confusion matrix has shown in Figure 4.10, where normal operation is called non-attack and Sparta SSH brute-force, MQTT brute-force ,aggressive scan, UDP scan are called attack.

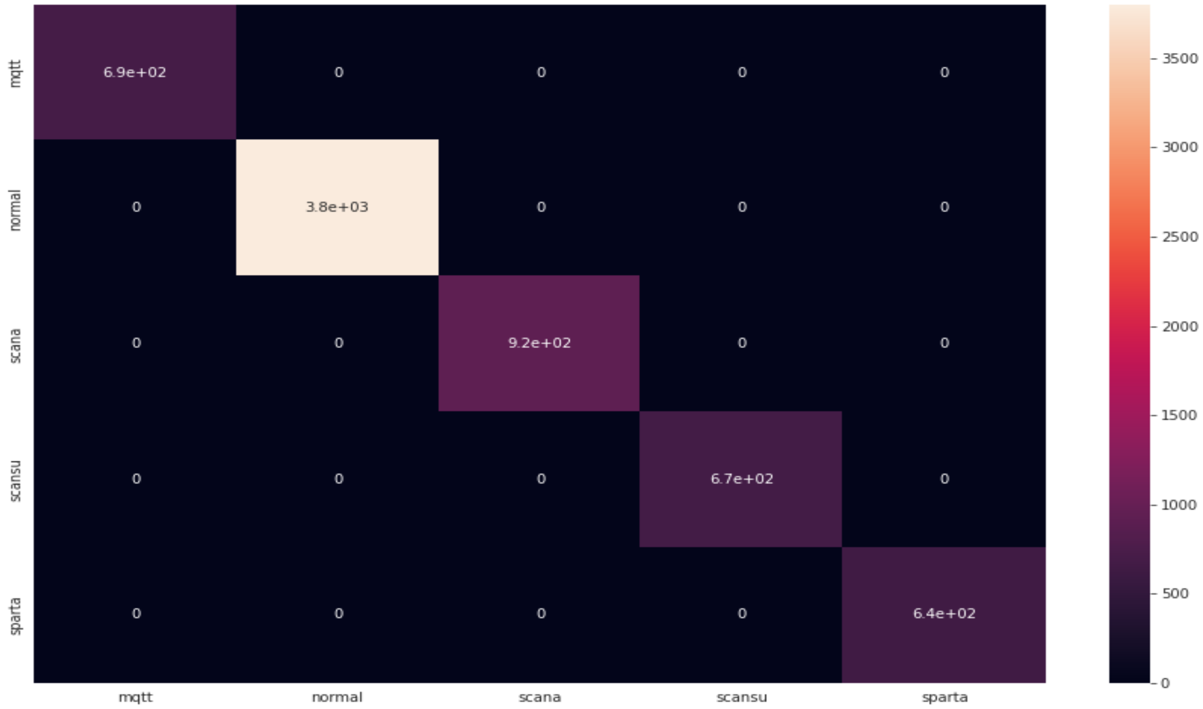


Figure 4.10: Confusion matrix

4.2.9 Epoch accuracy and epoch loss

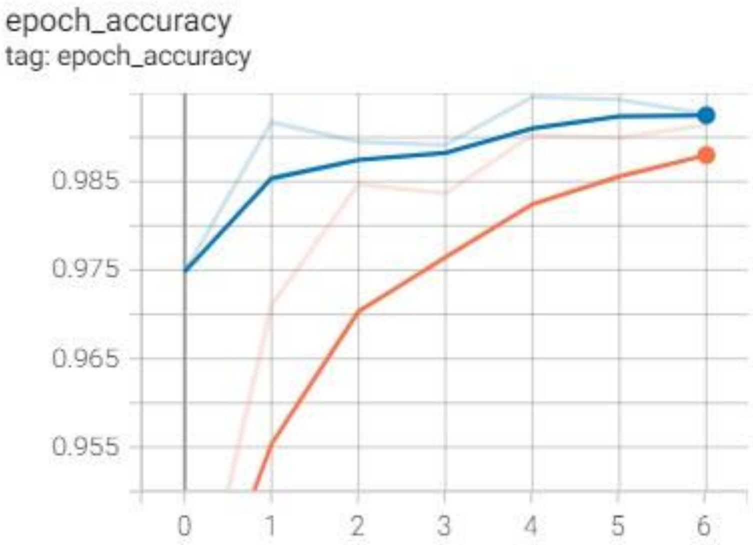


Figure 4.11: epoch accuracy

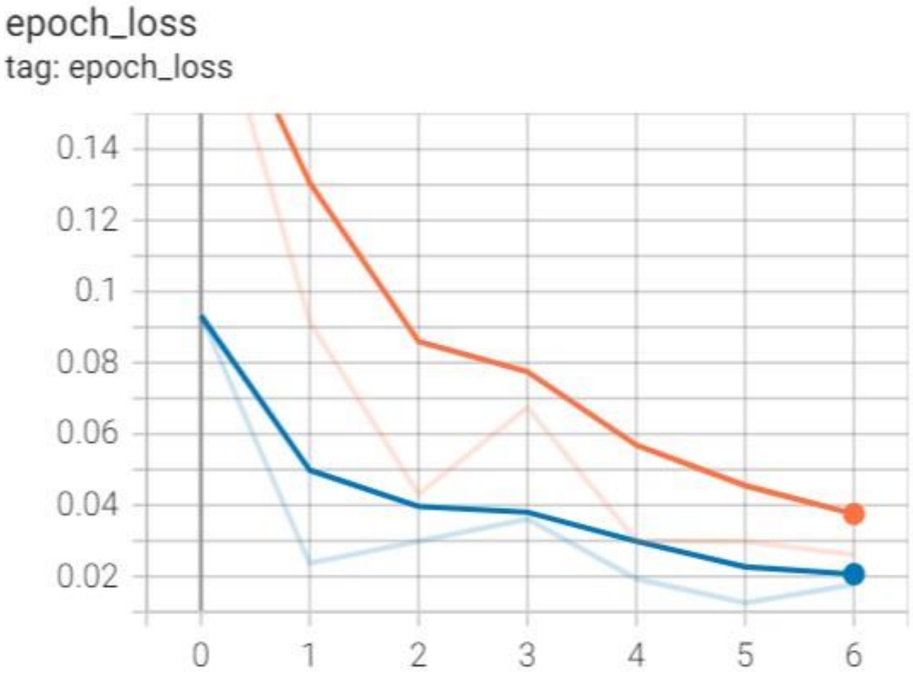


Figure 4.12: epoch loss

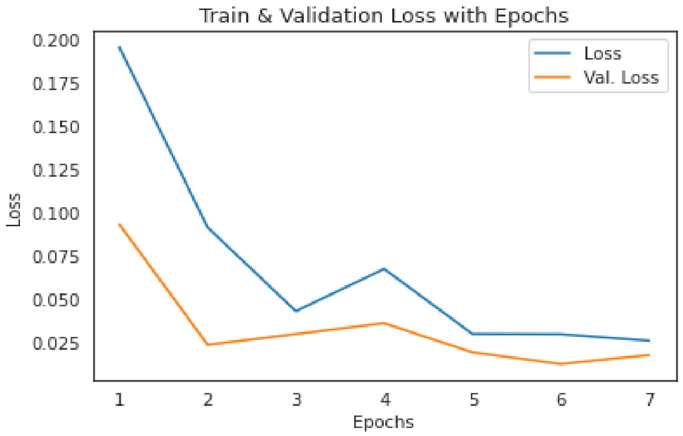


Figure 4.13: Train and validation loss with epochs

4.3 Summary

This chapter has presented the detailed performance evaluation of proposed model. Performance evaluation described in detail in the form of performance metrics. The proposed model outperformed the other models in both training and testing dataset.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Overview

The performance of proposed model RF-SVM-LSTM evaluates on google colab in terms of performance metrics such as Accuracy, Precision, Recall and F1-Score. The proposed scheme compared with different ML and DL models. It proves to be more effective model for detecting intrusion in IoT networks.

5.2 Conclusion

Attackers are using latest techniques to create the latest attacks like DoS, DDoS, U2R, R2L, Mirai, MiTM, Ransomware, Brute force, etc. to breach the security of IoT networks. In this study, a novel hybrid DL and ML based intrusion detection scheme is proposed. The main contribution of our proposed framework is the integration of RF, SVM, LSTM and Gradient Boosting algorithm that combines the benefits of respective strengths of ML and DL based Intrusion detection scheme to detect intrusion efficiently. We proposed generalized model that performed good on both train and test dataset by handling the problems of overfitting and underfittig. The new feature is generated using source IP and destination IP for checking if the data packet is from the same subnet or not. The data has pre-processed by using one hot encoding and normalization. In the proposed model hyperparameter tuning is applied for selecting the best hyperparameters and best epochs. The best features are identified for feature minimization. Moreover, the performance evaluation of proposed model is done by using MQTT-IOT-IDS 2020 dataset. Furthermore, performance of proposed model compared with different ML and DL algorithms such as SVM, LR, DNN and RNN. The proposed model outperformed other models in terms of higher accuracy, precision, recall and F1-Score. It shows that our proposed model can detect the intrusion in IoT based smart healthcare effectively.

5.3 Future Work

In future, we wish to apply the proposed model on other datasets to measure the effectiveness of detecting intrusion. Moreover, it should be applied in IoT based real time healthcare systems.

References

[1] S. Jüngling, J. Lutz, S. Korkut, and J. Jäger, *Business Information Systems and Technology 4.0*, vol. 141. Springer International Publishing, 2018. doi: 10.1007/978-3-319-74322-6.

[2] H. Harb, A. Mansour, A. Nasser, E. M. Cruz, and I. De La Torre Diez, “A Sensor-Based Data Analytics for Patient Monitoring in Connected Healthcare Applications,” *IEEE Sens. J.*, vol. 21, no. 2, pp. 974–984, 2021, doi: 10.1109/JSEN.2020.2977352.

[3] P. D. V Chandran, S. Adarkar, A. Joshi, and P. Kajbaje, “Digital Medicine : An android based application for health care system,” *Int. Res. J. Eng. Technol.*, vol. 4, no. 4, 2017, [Online]. Available: https://www.irjet.net/archives/V4/i4/IRJET-V4I4584.pdf

[4] S. M. Taghavinejad, M. Taghavinejad, L. Shahmiri, M. Zavvar, and M. H. Zavvar, “Intrusion Detection in IoT-Based Smart Grid Using Hybrid Decision Tree,” *2020 6th Int. Conf. Web Res. ICWR 2020*, pp. 152–156, 2020, doi: 10.1109/ICWR49608.2020.9122320.

[5] J. D. Lee, H. S. Cha, S. Rathore, and J. H. Park, “M-IDM: A multi-classification based intrusion detection model in healthcare iot,” *Comput. Mater. Contin.*, vol. 67, no. 2, pp. 1537–1553, 2021, doi: 10.32604/cmc.2021.014774.

[6] S. Anwar *et al.*, “From intrusion detection to an intrusion response system: Fundamentals, requirements, and future directions,” *Algorithms*, vol. 10, no. 2, 2017, doi: 10.3390/a10020039.

[7] M. Letafati, A. Kuhestani, K. K. Wong, and M. J. Piran, “A Lightweight Secure and Resilient Transmission Scheme for the Internet of Things in the Presence of a Hostile Jammer,” *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4373–4388, 2021, doi: 10.1109/JIOT.2020.3026475.

[8] M. Keshk, B. Turnbull, N. Moustafa, D. Vatsalan, and K. K. R. Choo, “A Privacy-Preserving-Framework-Based Blockchain and Deep Learning for Protecting Smart Power Networks,” *IEEE Trans. Ind. Informatics*, vol. 16, no. 8, pp. 5110–5118, 2020, doi: 10.1109/TII.2019.2957140.

[9] N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac, and P. Faruki, “Network Intrusion Detection for IoT Security Based on Learning Techniques,” *IEEE Commun. Surv. Tutorials*, vol. 21, no. 3, pp. 2671–2701, 2019, doi: 10.1109/COMST.2019.2896380.

[10] C. Yin, Y. Zhu, J. Fei, and X. He, “A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks,” *IEEE Access*, vol. 5, pp. 21954–21961, 2017, doi: 10.1109/ACCESS.2017.2762418.

[11] M. A. Khan, “HCRNNIDS : Hybrid Convolutional Recurrent Neural,” 2021.

[12] T. Cisco and A. Internet, “Cisco: 2020 CISO Benchmark Report,” *Comput. Fraud Secur.*, vol. 2020, no. 3, pp. 4–4, 2020, doi: 10.1016/s1361-3723(20)30026-9.

[13] A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, and A. Alazab, “Hybrid intrusion detection system based on the stacking ensemble of C5 decision tree classifier and one class support vector machine,” *Electron.*, vol. 9, no. 1, 2020, doi: 10.3390/electronics9010173.

[14] Y. K. Saheed and M. O. Arowolo, “Efficient Cyber Attack Detection on the Internet of Medical Things-Smart Environment Based on Deep Recurrent Neural Network and Machine Learning Algorithms,” *IEEE Access*, vol. 9, pp. 161546–161554, 2021, doi: 10.1109/ACCESS.2021.3128837.

[15] A. Aldaej, T. A. Ahanger, M. Atiquzzaman, I. Ullah, and M. Yousufudin, “Smart Cybersecurity Framework for IoT-Empowered Drones: Machine Learning Perspective,” *Sensors*, vol. 22, no. 7, pp. 1–25, 2022, doi: 10.3390/s22072630.

[16] Z. Liu, N. Thapa, A. Shaver, K. Roy, X. Yuan, and S. Khorsandroo, “Anomaly detection on lot network intrusion using machine learning,” *2020 Int. Conf. Artif. Intell. Big Data, Comput. Data Commun. Syst. icABCD 2020 - Proc.*, 2020, doi: 10.1109/icABCD49160.2020.9183842.

[17] S. Zahoor and R. N. Mir, “Resource management in pervasive Internet of Things: A survey,” *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 33, no. 8, pp. 921–935, 2021, doi: 10.1016/j.jksuci.2018.08.014.

[18] N. D. Lane *et al.*, “07460664,” *Inf. Process. Sens. Networks (IPSN), 2016 15th ACM/IEEE Int. Conf.*, no. 1, pp. 1–12, 2016.

[19] Kunal and M. Dua, “Machine Learning Approach to IDS: A Comprehensive Review,” *Proc. 3rd Int. Conf. Electron. Commun. Aerosp. Technol. ICECA 2019*, pp. 117–121, 2019, doi: 10.1109/ICECA.2019.8822120.

[20] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, “Feature selection and classification in multiple class datasets: An application to KDD Cup 99 dataset,” *Expert Syst. Appl.*, vol. 38, no. 5, pp. 5947–5957, 2011, doi: 10.1016/j.eswa.2010.11.028.

[21] E. Viegas, A. O. Santin, A. Franca, R. Jasinski, V. A. Pedroni, and L. S. Oliveira, “Towards an energy-efficient anomaly-based intrusion detection engine for embedded systems,” *IEEE Trans. Comput.*, vol. 66, no. 1, pp. 163–177, 2017, doi: 10.1109/TC.2016.2560839.

[22] Y. Xin *et al.*, “Machine Learning and Deep Learning Methods for Cybersecurity,” *IEEE Access*, vol. 6, pp. 35365–35381, 2018, doi: 10.1109/ACCESS.2018.2836950.

[23] R. Swischuk, L. Mainini, B. Peherstorfer, and K. Willcox, “Projection-based model reduction: Formulations for physics-based machine learning,” *Comput. Fluids*, vol. 179, pp. 704–717, 2019, doi: 10.1016/j.compfluid.2018.07.021.

[24] A. H. A and K. Sundarakantham, “Machine Learning Based Intrusion,” *2019 3rd Int. Conf. Trends Electron. Informatics*, no. Icoei, pp. 916–920, 2019.

[25] H. K. Jabbar and R. Z. Khan, “Methods to Avoid Over-Fitting and Under-Fitting in Supervised Machine Learning (Comparative Study),” no. February, pp. 163–172, 2015, doi: 10.3850/978-981-09-5247-1_017.

[26] R. Sommer and V. Paxson, “Outside the closed world: On using machine learning for network intrusion detection,” *Proc. - IEEE Symp. Secur. Priv.*, pp. 305–316, 2010, doi: 10.1109/SP.2010.25.

[27] I. Ullah, A. Ullah, and M. Sajjad, “Towards a Hybrid Deep Learning Model for Anomalous Activities Detection in Internet of Things Networks,” *IoT*, vol. 2, no. 3, pp. 428–448, 2021, doi: 10.3390/iot2030022.

[28] Y. Zhang, P. Li, and X. Wang, “Intrusion Detection for IoT Based on Improved Genetic Algorithm and Deep Belief Network,” *IEEE Access*, vol. 7, pp. 31711–31722, 2019, doi: 10.1109/ACCESS.2019.2903723.

[29] Y. Liu, Y. Wang, and J. Zhang, “New machine learning algorithm: Random forest,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7473 LNCS, pp. 246–252, 2012, doi: 10.1007/978-3-642-34062-8_32.

[30] N. Farnaaz and M. A. Jabbar, “Random Forest Modeling for Network Intrusion Detection System,” *Procedia Comput. Sci.*, vol. 89, pp. 213–217, 2016, doi: 10.1016/j.procs.2016.06.047.

[31] S. Hanif, T. Ilyas, and M. Zeeshan, “Intrusion Detection in IoT Using Artificial Neural Networks on UNSW-15 Dataset,” *HONET-ICT 2019 - IEEE 16th Int. Conf. Smart Cities Improv. Qual. Life using ICT, IoT AI*, pp. 152–156, 2019, doi: 10.1109/HONET.2019.8908122.

[32] T. A. Mohamed, T. Otsuka, and T. Ito, *Towards machine learning based IoT intrusion detection service*, vol. 10868 LNAI. Springer International Publishing, 2018. doi: 10.1007/978-3-319-92058-0_56.

[33] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," *2015 Mil. Commun. Inf. Syst. Conf. MilCIS 2015 - Proc.*, 2015, doi: 10.1109/MilCIS.2015.7348942.

[34] M. Mohammadi *et al.*, "A comprehensive survey and taxonomy of the SVM-based intrusion detection systems," *J. Netw. Comput. Appl.*, vol. 178, no. December 2020, p. 102983, 2021, doi: 10.1016/j.jnca.2021.102983.

[35] H. Wang, J. Gu, and S. Wang, "An effective intrusion detection framework based on SVM with feature augmentation," *Knowledge-Based Syst.*, vol. 136, pp. 130–139, 2017, doi: 10.1016/j.knosys.2017.09.014.

[36] D. Jing and H. B. Chen, "SVM based network intrusion detection for the UNSW-NB15 dataset," *Proc. Int. Conf. ASIC*, pp. 1–4, 2019, doi: 10.1109/ASICON47005.2019.8983598.

[37] B. Ingre, A. Yadav, and A. K. Soni, "Decision tree based intrusion detection system for NSL-KDD dataset," *Smart Innov. Syst. Technol.*, vol. 84, no. Ictis 2017, pp. 207–218, 2018, doi: 10.1007/978-3-319-63645-0_23.

[38] R. Guzmán-Cabrera, B. P. Sánchez, T. P. Mukhopadhyay, J. M. L. García, and T. Cordova-Fraga, "Classification of opinions in cross domains involving emotive values," *J. Intell. Fuzzy Syst.*, vol. 36, no. 5, pp. 4877–4887, 2019, doi: 10.3233/JIFS-179035.

[39] N. A. Singh, J. Singh, and T. De, "Distributed denial of service attack detection using naive bayes classifier through info gain feature selection," *ACM Int. Conf. Proceeding Ser.*, vol. 25-26-Augu, no. Icimia, pp. 711–717, 2016, doi: 10.1145/2980258.2980379.

[40] B. Zhang, Z. Liu, Y. Jia, J. Ren, and X. Zhao, “Zhang, B., Liu, Z., Jia, Y., Ren, J., & Zhao, X. (2018). Network Intrusion Detection Method Based on PCA and Bayes Algorithm. In Security and Communication Networks (Vol. 2018). https://doi.org/10.1155/2018/1914980Network Intrusion Detection Method Based ,” *Secur. Commun. Networks*, vol. 2018, 2018.

[41] P. Lavanya, A. Sangeetha, and S. Krishnan, “Intrusion detection using machine learning,” *Int. J. Recent Technol. Eng.*, vol. 8, no. 2 Special Issue 6, pp. 832–837, 2019, doi: 10.35940/ijrte.B1154.0782S619.

[42] L. Li, H. Zhang, H. Peng, and Y. Yang, “Nearest neighbors based density peaks approach to intrusion detection,” *Chaos, Solitons and Fractals*, vol. 110, pp. 33–40, 2018, doi: 10.1016/j.chaos.2018.03.010.

[43] S. S. Swarna Sugi and S. R. Ratna, “Investigation of machine learning techniques in intrusion detection system for IoT network,” *Proc. 3rd Int. Conf. Intell. Sustain. Syst. ICISS 2020*, pp. 1164–1167, 2020, doi: 10.1109/ICISS49785.2020.9315900.

[44] H. H. Pajouh, R. Javidan, R. Khayami, A. Dehghantanha, and K. K. R. Choo, “A Two-Layer Dimension Reduction and Two-Tier Classification Model for Anomaly-Based Intrusion Detection in IoT Backbone Networks,” *IEEE Trans. Emerg. Top. Comput.*, vol. 7, no. 2, pp. 314–323, 2019, doi: 10.1109/TETC.2016.2633228.

[45] F. Nelli, “Machine Learning with scikit-learn,” *Python Data Anal.*, pp. 237–264, 2015, doi: 10.1007/978-1-4842-0958-5_8.

[46] M. Hasan, M. M. Islam, M. I. I. Zarif, and M. M. A. Hashem, “Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches,” *Internet of Things (Netherlands)*, vol. 7, p. 100059, 2019, doi: 10.1016/j.iot.2019.100059.

[47] S. Waskle, L. Parashar, and U. Singh, “Intrusion Detection System Using PCA with Random Forest Approach,” *Proc. Int. Conf. Electron. Sustain. Commun. Syst. ICESC 2020*, no. Icesc, pp. 803–808, 2020, doi: 10.1109/ICESC48915.2020.9155656.

[48] P. Shukla, “ML-IDS: A machine learning approach to detect wormhole attacks in Internet of Things,” *2017 Intell. Syst. Conf. IntelliSys 2017*, vol. 2018-Janua, no. September, pp. 234–240, 2018, doi: 10.1109/IntelliSys.2017.8324298.

[49] V. Jain and M. Agrawal, “Applying Genetic Algorithm in Intrusion Detection System of IoT Applications,” *Proc. 4th Int. Conf. Trends Electron. Informatics, ICOEI 2020*, no. Icoei, pp. 284–287, 2020, doi: 10.1109/ICOEI48184.2020.9143019.

[50] C. Azad and V. K. Jha, *Decision tree and genetic algorithm based intrusion detection system*, vol. 476, no. Mccs 2017. Springer Singapore, 2019. doi: 10.1007/978-981-10-8234-4_13.

[51] A. A. Aburomman and M. Bin Ibne Reaz, “A novel SVM-kNN-PSO ensemble method for intrusion detection system,” *Appl. Soft Comput. J.*, vol. 38, pp. 360–372, 2016, doi: 10.1016/j.asoc.2015.10.011.

[52] A. Mariot, S. Sgoifo, and M. Sauli, “I gozzi endotoracici: contributo casistico-clinico (20 casi),” *Friuli Med.*, vol. 19, no. 6, 1964.

[53] A. Shaver, Z. Liu, N. Thapa, K. Roy, B. Gokaraju, and X. Yuan, “Anomaly based intrusion detection for iot with machine learning,” *Proc. - Appl. Imag. Pattern Recognit. Work.*, vol. 2020-Octob, pp. 4–9, 2020, doi: 10.1109/AIPR50011.2020.9425199.

[54] A. Verma and V. Ranga, “Machine Learning Based Intrusion Detection Systems for IoT Applications,” *Wirel. Pers. Commun.*, vol. 111, no. 4, pp. 2287–2310, 2020, doi: 10.1007/s11277-019-06986-8.

[55] P. Van Huong, L. D. Thuan, L. T. Hong Van, and D. V. Hung, “Intrusion detection in IoT systems based on deep learning using convolutional neural network,” *Proc. - 2019 6th NAFOSTED Conf. Inf. Comput. Sci. NICS 2019*, pp. 448–453, 2019, doi: 10.1109/NICS48868.2019.9023871.

[56] I. Idrissi, M. Boukabous, M. Azizi, O. Moussaoui, and H. El Fadili, “Toward a deep learning-based intrusion detection system for iot against botnet attacks,” *IAES Int. J. Artif. Intell.*, vol. 10, no. 1, pp. 110–120, 2021, doi: 10.11591/ijai.v10.i1.pp110-120.

[57] H. Zhang, C. Q. Wu, S. Gao, Z. Wang, Y. Xu, and Y. Liu, “An Effective Deep Learning Based Scheme for Network Intrusion Detection,” *Proc. - Int. Conf. Pattern Recognit.*, vol. 2018-Augus, pp. 682–687, 2018, doi: 10.1109/ICPR.2018.8546162.

[58] S. Wankhede and D. Kshirsagar, “DoS Attack Detection Using Machine Learning and Neural Network,” *Proc. - 2018 4th Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2018*, 2018, doi: 10.1109/ICCUBEA.2018.8697702.

[59] M. A. Khan *et al.*, “A deep learning-based intrusion detection system for mqtt enabled iot,” *Sensors*, vol. 21, no. 21, pp. 1–25, 2021, doi: 10.3390/s21217016.

[60] “D9Aef4F86F019Dca209Aee1Cc6805B0545C68a08 @ Joseaveleira.Es.” [Online]. Available: https://joseaveleira.es/dataset

[61] A. Nagisetty and G. P. Gupta, “Framework for detection of malicious activities in IoT networks using keras deep learning library,” *Proc. 3rd Int. Conf. Comput. Methodol. Commun. ICCMC 2019*, no. Iccmc, pp. 633–637, 2019, doi: 10.1109/ICCMC.2019.8819688.

[62] Y. Xu, Y. Tang, and Q. Yang, “Deep learning for IoT intrusion detection based on LSTMs-AE,” *ACM Int. Conf. Proceeding Ser.*, pp. 64–68, 2020, doi: 10.1145/3421766.3421891.

[63] Y. Imrana, Y. Xiang, L. Ali, and Z. Abdul-Rauf, “A bidirectional LSTM deep learning approach for intrusion detection,” *Expert Syst. Appl.*, vol. 185, no. June, p. 115524, 2021, doi: 10.1016/j.eswa.2021.115524.

[64] X. Fu, N. Zhou, L. Jiao, H. Li, and J. Zhang, “The robust deep learning–based schemes

for intrusion detection in Internet of Things environments," *Ann. des Telecommun. Telecommun.*, vol. 76, no. 5–6, pp. 273–285, 2021, doi: 10.1007/s12243-021-00854-y.

[65] E. Almazrouei, G. Gianini, C. Mio, N. Almoosa, and E. Damiani, "Using autoencoders for radio signal denoising," *Q2SWinet 2019 - Proc. 15th ACM Int. Symp. QoS Secur. Wirel. Mob. Networks*, pp. 11–17, 2019, doi: 10.1145/3345837.3355949.

[66] Z. Chen, C. K. Yeo, B. S. Lee, and C. T. Lau, "基于自动编码器的网络异常检测," *Wirel. Telecommun. Symp.*, vol. 2018-April, pp. 1–5, 2018.

[67] Y. Li, P. Gao, and Z. Wu, "Intrusion Detection Method Based on Sparse Autoencoder," *2021 3rd Int. Conf. Comput. Commun. Internet, ICCCI 2021*, pp. 63–68, 2021, doi: 10.1109/ICCCI51764.2021.9486776.

[68] V. Upadhya and P. S. Sastry, "An Overview of Restricted Boltzmann Machines," *J. Indian Inst. Sci.*, vol. 99, no. 2, pp. 225–236, 2019, doi: 10.1007/s41745-019-0102-z.

[69] A. Dawoud, S. Shahristani, and C. Raun, "Deep learning and software-defined networks: Towards secure IoT architecture," *Internet of Things (Netherlands)*, vol. 3–4, pp. 82–89, 2018, doi: 10.1016/j.iot.2018.09.003.

[70] S. Teng, N. Wu, H. Zhu, L. Teng, and W. Zhang, "SVM-DT-based adaptive and collaborative intrusion detection," *IEEE/CAA J. Autom. Sin.*, vol. 5, no. 1, pp. 108–118, 2018, doi: 10.1109/JAS.2017.7510730.

[71] M. Roopak, G. Yun Tian, and J. Chambers, "Deep learning models for cyber security in IoT networks," *2019 IEEE 9th Annu. Comput. Commun. Work. Conf. CCWC 2019*, pp. 452–457, 2019, doi: 10.1109/CCWC.2019.8666588.

[72] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," *J. Inf. Secur. Appl.*, vol. 50, p. 102419, 2020, doi: 10.1016/j.jisa.2019.102419.

[73] "MQTT-IoT-IDS2020: MQTT Internet of Things Intrusion Detection Dataset | IEEE DataPort." https://ieee-dataport.org/open-access/mqtt-iot-ids2020-mqtt-internet-things-intrusion-detection-dataset (accessed Aug. 26, 2022).

[74] "MQTT-IoT-IDS2020: MQTT Internet of Things Intrusion Detection Dataset | IEEE DataPort."

[75] G. Y. Lee, L. Alzamil, B. Doskenov, and A. Termehchy, "A Survey on Data Cleaning Methods for Improved Machine Learning Model Performance," no. September, pp. 0–6, 2021.

[76] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. February 2017, pp. 321–357, 2002, doi: 10.1613/jair.953.

[77] "Feature extraction: A survey | IEEE Journals & Magazine | IEEE Xplore."

[78] D. Shah, Z. Y. Xue, and T. M. Aamodt, "L e r n," no. Section 2, pp. 1–35, 2022.

[79] C. Seger, "An investigation of categorical variable encoding techniques in machine learning: binary versus one-hot and feature hashing," *Degree Proj. Technol.*, p. 41, 2018.

[80] S. J. Malebary and Y. D. Khan, "Evaluating machine learning methodologies for identification of cancer driver genes," *Sci. Rep.*, vol. 11, no. 1, pp. 1–14, 2021, doi: 10.1038/s41598-021-91656-8.

[81] P. A. A. Resende and A. C. Drummond, "A survey of random forest based methods for intrusion detection systems," *ACM Comput. Surv.*, vol. 51, no. 3, 2018, doi: 10.1145/3178582.

[82] M. Schonlau and R. Y. Zou, "The random forest algorithm for statistical learning,"

Stata J., vol. 20, no. 1, pp. 3–29, 2020, doi: 10.1177/1536867X20909688.

[83] M. Awad and R. Khanna, “Efficient learning machines: Theories, concepts, and applications for engineers and system designers,” *Effic. Learn. Mach. Theor. Concepts, Appl. Eng. Syst. Des.*, no. July 2018, pp. 1–248, 2015, doi: 10.1007/978-1-4302-5990-9.

[84] D. Rengasamy, M. Jafari, B. Rothwell, X. Chen, and G. P. Figueredo, “Deep learning with dynamically weighted loss function for sensor-based prognostics and health management,” *Sensors (Switzerland)*, vol. 20, no. 3, 2020, doi: 10.3390/s20030723.