National College of Ireland

Intrusion Detection in IoT devices using Zero Bias DNN

MSc Research Project
Cyber Security

Aksa Anna Shajan
Student ID: 20178875

School of Computing
National College of Ireland

Supervisor: Dr. Vanessa Ayala-Rivera

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: ……AKSA ANNA SHAJAN……

Student ID: …20178875……..……

Programme: ……Cyber Security…… **Year:** …2021-2022…..

Module: ……MSc Research Project……..……

Supervisor: …… Dr. Vanessa Ayala-Rivera……..……

Submission Due Date: ……16 Dec 2021……..……

Project Title: …Intrusion detection in IoT devices using Zero bias DNN……..……

Word Count: ……5097…… **Page Count**……18……..……..

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: …Aksa Anna Shajan……

Date: …16 Dec 2021……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Intrusion Detection in IoT devices using Zero Bias DNN

Aksa Anna Shajan
20178875

Abstract

By providing smart services in a way that would otherwise be inconceivable, the Internet of Things has become an unavoidable aspect of today's life. IoT devices generally have only one privilege level, both externally and internally, so once this privilege is achieved, it becomes easy for attackers to exploit the vulnerabilities. Because of the huge demand and privilege weaknesses, attackers aim to exploit flaws to launch attacks like DDoS attacks, so ensuring security becomes difficult. It's vital to identify malicious activities effectively. Insecure IoT devices can cause a tremendous impact in a variety of ways. Among the various techniques used for intrusion detection, machine learning approaches proves to be best with better results. Developing an efficient and effective intrusion detection system is challenging as attackers come with a variety of attack patterns each day. This study proposes a new machine learning method - zero bias deep neural network for identifying intrusions in IoT devices. A dataset named MQTT-IoT IDS 2020, a latest dataset on IoT normal and attack patterns, from IEEEPort, is used in our model. The previous research papers on IoT IDS have used datasets that are used in all general intrusion cases of network attacks, not particularly for IoT. SMOTE oversampling technique has been used to resolve the data imbalance and Random Forest classifier algorithm was used for feature elimination. Our experiment shows that zero bias DNN produces a satisfactory result with an accuracy of 92% on our dataset.

1 Introduction

The IoT devices or Internet of things are smart devices developed for network services. They have sensors inbuilt in them. These devices can process data when connected to the internet. IoT devices are not designed for security purposes instead they were developed for making the impossible to possible. IoT devices are widely used nowadays because of their efficiency. The number of IoT devices has increased dramatically, from 15 billion in 2015 to 46 million by the end of 2021. By 2025, it is estimated to have 80 million IoT devices. Wireless data transmission is possible with the help of IoT devices. There is a rapid increase in the number of IoT devices being used. The advancement in technology is a boon but can be a bane if it’s not secure. Confidentiality, integrity, and availability are considered as the backbone of security. Data security will be possible only if all three factors are unaffected. A large amount of data is being transferred or handled among the IoT devices, since these devices are not developed for security, there will be vulnerabilities existing in IoT devices that can be utilized by attackers to perform attacks. Attacks can be stealing information to perform huge, distributed denial of service attacks by using the IoT devices as botnets. The impact of IoT

attacks can be too high than the expectation, so it is very important to have a proper intrusion detection system.

Attackers usually focus on the network layer to perform the attacks as the communication between the IoT devices and server occurs in the network layer, attackers could perform IP spoofing, sniffing, and so on in this layer. [4] IoT systems can be accessed from anywhere on the planet. Critical changes to pre-existing security standards for information and wireless networks should be made to achieve quality IoT security solutions.

The purpose of this paper is to develop an effective intrusion detection system for network layer intrusions. Detecting any form of scanning or intrusions attempting to take place in the communication channel can help protect the device from further attacks. Traditional intrusion detection methods are less successful than deep learning alternatives. Deep learning systems use numerous hidden layers or processing layers instead of a single processing or hidden layer in a traditional neural network. We presented zero bias deep neural networks for IoT detection to be more accurate and efficient. Yongxin Liu et al. [5] were the first to introduce zero bias DNN for IoT device recognition.

Research Question:
How effective is the Zero bias DNN for intrusion detection using the MQTT-IoT-ISD2020 dataset?

The traditional methods of intrusion detection were based on rule-based IDS. Those techniques seemed to be less effective. The modern methods of machine learning had produced better results, among which DNN proves to be the best. In the paper [5] zero bias DNN method has been used for proper identification of IoT devices and proved to have high accuracy. IoT devices use the MQTT protocol for communication. MQTT-IoT-IDS2020is one of the latest datasets which are completely based on IoT devices. Using the latest dataset with new this robust method could produce a satisfactory result.

The Random Forest Classifier is the algorithm for our new Zero Bias DNN model. The main issue with the previous papers on this topic was that all the papers had used old datasets for training the model. We have used a reliable dataset for IoT intrusions but the challenge that was faced during the current approach was the issue of accuracy. The main contribution of this model is creating a new method- Zero bias DNN along with training by the latest dataset. This paper contains five sections. Section II provides the literature review on related topics, Section III discusses the research methodology, section IV explains the design specifications and section V explains the implementation, section VI provides the evaluation, and finally, section VII concludes the paper.

2 Related Work

Any activity that compromises the data's confidentiality, integrity, or availability can render the device or system vulnerable. Dos attacks, Botnets, Identity theft, and ransomware attacks are some of the most common threats against IoT devices. The purpose of a dos attack is to make a service inaccessible by flooding the server with requests and bringing it down. Botnet attacks are carried out by turning IoT devices into bots that carry out DDoS and Dos attacks. Among the IoT attacks, the Botnet attack raised a serious security issue for the internet of

things. Because of the risk of losing confidentiality, integrity, or availability, as well as the worry of losing customer trust, the Mirai DDoS attack has caused widespread concern in the IT industry. Mirai looked for machines with open telnet ports and took advantage of the flaw by uploading malware[6][7].

IoT technology [9]: Figure 1 depicts the technologies that are used to power an IoT device. Sensors, radio-frequency identification technology, nanotechnologies, and smart technologies are all part of the Internet of Things. RFID stands for Radio Frequency Identification, and it is a microchip that is used to identify objects wirelessly. Sensors are used to collect data and determine the physical condition of items. Smart technology devices, such as smartwatches, adapt smart services while accessing resources in the IoT system, boosting network processing capacity. Nanotechnology aids in the development of intelligent solutions.

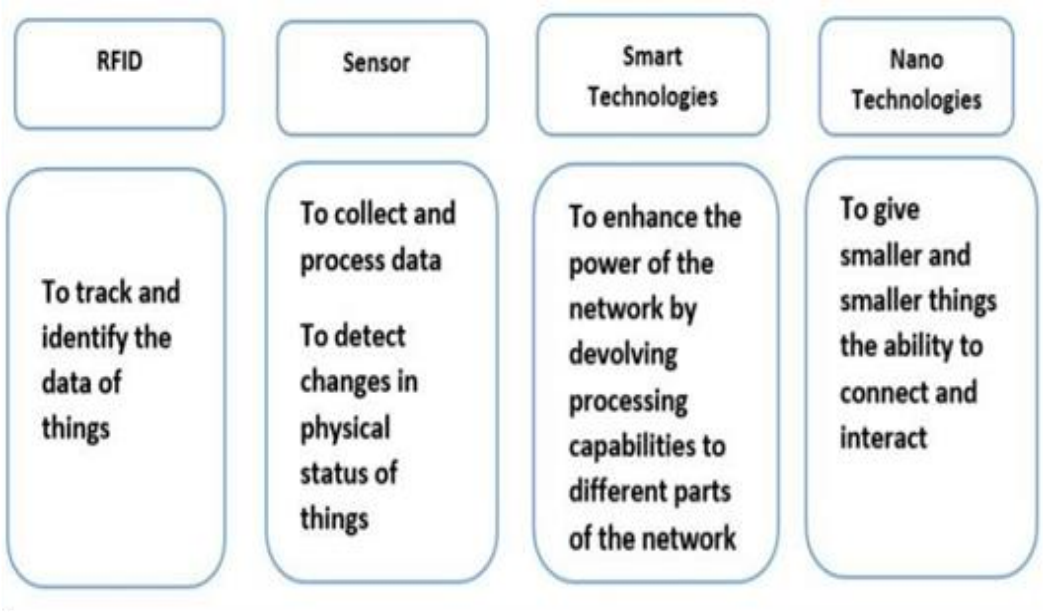


Figure 1

Machine learning techniques:

The traditional intrusion detection systems were rule-based IDS, which means the detection was done in a way that the system searches for particular commands or keywords. Those IDSs were not satisfactory. So there was a need to have new methods for intrusion detection. Machine learning is a field of artificial intelligence based on the premise that computers can learn from data, recognize patterns, and make judgments with little or no human involvement. A Machine learning IDS tries to learn the normal and attack types of network traffic generated by IoT devices, and then discover abnormalities based on algorithms and deviations from those normal types of traffic. Some of the machine learning techniques used by researchers for intrusion detection are explained below.

Artificial neural networks (ANN)

It is also known as deep learning, which use a set of data inputs, values, and assumptions to try to replicate the human brain and work together to effectively detect, categorize, and characterize items in the data. The ANN's neurons are utilized to create complicated

hypotheses; the more neurons in the ANN, the more complex the hypotheses become. The assumptions are evaluated by using a feedback mechanism to set the input nodes, and the event streams are propagated through the network to the output, where they are classed as normal or compromised. Gradient descents are employed at this point to send the error in the output node back through the network via a backpropagation process, allowing the error in the hidden nodes to be estimated. As a result, the cost - function's gradient can be determined. To learn the pattern established by the system, the neural network system is trained.

In 2016, Elike et.al. [10] employed the artificial neural network (ANN) approach to do threat analysis in IoT networks. In order to identify a DoS attack on the network, an ANN is used as an offline Intrusion Detection System, collecting and analyzing data from many parts of the IoT network. The network was trained with 2313 samples, then verified with 496 samples and tested with 496 samples. The detection was carried out by categorizing normal and malicious behavior. The intrusion detection system was successful in recognizing DoS and DDoS patterns, but it was unsuccessful in detecting other types of attacks.
In 2018, Moustafa et. al. [12], proposed an Adaboost learning approach that integrates three machine learning models – ANN, tree, and Naive Bayes – to create a system that is efficient. The experiment utilized data from the NIMS botnet as well as UNSW-NB15. The suggested system is a network intrusion detection system in the most basic sense. For the detection of malicious actions, TCP/IP layer features are extracted. Using the Adaboost method, this strategy does a comprehensive examination. Bro-IDS was used to extract features from protocols like HTTP, MQTT, and DNS to create a useful solution. The proposed system exhibited a high rate of detection with a low rate of false positives. The approach merely looked at network traffic and didn't catch zero-day assaults.

RNNs (recurrent neural networks)

They are a type of neural network that can be used to model sequence data. Other algorithms can't create prediction outcomes with sequential data like recurrent neural networks can. RNNs are a sort of neural network that is both powerful and robust, and they are one of the most promising algorithms in use because they are the only ones having an internal memory.

In 2017, the Gated-Recurrent-Unit (GRU) algorithm of RNN was employed by Manoj et. Al [9] for intrusion detection utilizing the recurrent neural networks model RNN. The experiment employed 21% of the KDD 99'Cup intrusion detection dataset to determine the system's accuracy. The system was only successful in detecting infiltration at the TCP/IP layer, and the GRU did not function well. The study proved unable to deal with large amounts of real-time IoT data.

The Bi-directional Long Short-Term Memory Recurrent Neural Network (BLSTM RNN) was introduced in 2019 by Roy et. al.[15], as a new solution for detecting assaults in the IoT network. The new model was created using the Keras deep learning framework and the Google TensorFlow library. UNSW-NB15, the same dataset as [13],[12], is used here as well. The method is a multi-layer deep learning algorithm that achieved 95 percent accurate results. Five different types of attacks were detected by the system. Even though it boasts a 95% accuracy rate, the results were not satisfactory.

Deep neural network (DNN)

It is also known as a deep net, which is a neural network with a high level of complexity, usually at least two layers. The ANN has only one hidden layer whereas when the complexity increases by increasing the number of hidden layers, the DNN is formed.
In 2017, [11] a study of the internet of things network was carried out by classifying the assaults using the DNN technique, with an emphasis on the feed-forward architecture. The experiment used three datasets: UNSW-NB15, CIDDS-001, and GPRS. In terms of detecting attacks on wireless networks, DNN performed admirably. The DNN model did not perform a good job of detecting anomalies. There were also a number of categorization difficulties, including unbalanced datasets and skewed results.
To identify intrusions in industrial IoTs, a novel network intrusion detection system was introduced in 2018 by Hawareh et. al. [13]. A deep auto-encoder and a deep feedforward neural network were employed in the detection technique. The information from the TCP/IP packets was used to validate the data. The datasets utilized to train the system were NSL-KDD and UNSW-NB15. The system is a high-accuracy anomaly detection system with a low false-positive rate.
The DNN approach was applied in conjunction with blockchain in 2019 [18]. NSL-KDD was the dataset used. The model is a multi-agent system, which means it may be employed in a variety of IoT environments of various sizes. To make the system more secure, the communication agent's operations will be recorded on the blockchain. The system's performance has improved as a result of the introduction of multi-agents. The experiment's outcome had high accuracy, although the system missed detecting several rare types of attacks.

Convolution neural networks (CNN)

They are a type of feed-forward neural network that has a deep structure and performs convolution calculations. It has representation learning capabilities and can classify input data based on its hierarchical structure. The convolutional neural network is one of the most well-known neural networks in deep learning technology, and it has achieved numerous breakthroughs in image interpretation and processing.
A new feature selection technique is implemented in 2021 by Parimala et. al. [3]. The conditional random field (CRF) and spider monkey optimization algorithms are combined in this algorithm (SMO). The combined algorithm aids in the extraction of the dataset's significant features. To begin, the CRF is used to select the features that have been contributed. The SMO is then used in the reduced features dataset to refine the most important characteristics. The classification is performed using the Convolutional Neural Network CNN approach. Feature selection did not function as well as expected.
Janani et. al. [21] in their paper examines the performance of various deep learning algorithms for intrusion detection. The detection accuracy, precision, false-positive rate, and true positive rate of the Decision Tree Classifier (DT), Deep Neural Network (DNN), and Convolutional Neural Network (CNN) models were evaluated. The DNN model performs better than the other two strategies, according to the analysis report[22].

Datasets:

Any machine learning model would be incomplete without data. Although a dataset comprises many different bits of data, it can be used to train an algorithm with the purpose of identifying predictable patterns within the dataset as a whole. One can evaluate trends and hidden patterns and make judgments based on the dataset.

KDD 99'Cup intrusion detection dataset, NSL-KDD, UNSW-NB15, CIDDS-001, and GPRS are some of the datasets previously used by researchers for intrusion detection systems. The KDD 99'cup is a dataset from the military network which contains a large variety of simulated intrusions from the network. NSL-KDD is a data set designed to address some of the issues with the KDD'99 data set. The NSL-KDD train and test sets have a reasonable amount of records. The UNSW-NB15 has modern attack patterns, normal modern traffic patterns, and diverse sets for training and testing. UNSW-NB15 has 42 attributes and has 31.94v percent normal and 68.06 percent, malicious classes. CIDDS-001 is a dataset for intrusion detection systems that use anomaly detection. GPRS is a wireless system dataset. All the above-mentioned datasets are general datasets that contain intrusions on the network that includes not only IoT intrusions but also many other intrusions that happen in the network layer. The BoT-IoT, IoT Network Intrusion, MQTT-IoT-IDS2020, and IoT-23 intrusion detection datasets, were the datasets related to IoT, described in a study conducted by Norman et.al.[23]. In order to have an effective, reliable, and up-to-date intrusion detection system, the detection system should be trained with datasets that contain a modern set of data connected to IoT devices and IoT traffics.

In the Internet of Things, the Message Queuing Telemetry Transport (MQTT) protocol is one of the most extensively used standards for machine-to-machine communication (IoT). MQTT-IoT-IDS2020 is the first dataset to model an MQTT-based network. The dataset was created using a simulated MQTT network architecture. There are five files in the dataset- MQTT bruteforce.csv, normal.csv, Scan A.csv, Scan sU.csv, and sparta.csv. MQTT-IoT-IDS2020 seems to be a good dataset for training an intrusion detection system for IoT.

The Zero Bias DNN

The zero-bias DNN was first used in 2020 as a solution for accurate IoT device identification using Zero-Bias deep learning. In the DNN architecture, the convolutional layers with skip connections were used to extract latent features, followed by a dense layer and a softmax layer for final classification. The last dense layer took a different method by replacing a neural network's last dense layer with a zero-bias layer.

The project was confronted with two types of difficulties. The interpretation of neural network operations is a difficulty, as is reporting unknown devices using a deep neural network. For DNNs to validate current devices and detect unfamiliar devices, the technique produced a zero-bias dense layer. The proposed architecture has been proved to be effective in managing large-signal recognition and improving convolutional neural network performance. The Zero-Bias layer improved interpretability and made it easier to report unknown devices.

Having a thorough study about IoT devices, machine learning techniques, and datasets, we have identified the following:

- Deep neural networks are a more efficient and commonly used technique for intrusion detection.

- No research paper has used a dataset completely based on IoT data for intrusion detection, MQTT-IoT-IDS 2020 is the latest and most suitable dataset for training an IoT IDS.
- Using a Zero bias layer in DNN makes the system more robust.

3 Research Methodology

The IoT market was valued at $190 billion in 2018, and it is expected to reach $1102.6 billion by 2022. The market is expected to increase at a compound annual rate of 24.7 percent (CAGR) by 2028. The Internet of Things' unique traits has also brought with them a slew of new security and privacy issues, which are a critical concern for IoT adoption's long-term viability. The Internet of Things is commonly used by attackers to perform several cyber-attacks because it is a centralized device. Over the past few years, the number of cyber-attacks has been increasing. When security became a concern, deep learning techniques were developed for identifying intrusion and detection in IoT devices. We have used a zero-bias deep learning technique for intrusion detection in IoT devices.

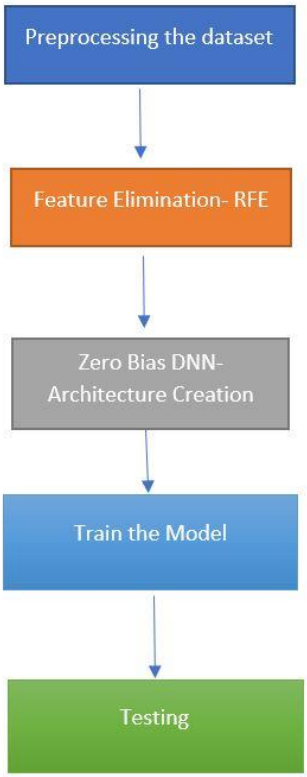


Figure 2

As illustrated in Figure 2, our approach consists of five steps. Pre-processing is the first step in developing our model. The Message Queuing Telemetry Transport (MQTT) protocol is one of the most widely used standards for machine-to-machine communication in the Internet of Things (IoT). The growing number of IoT devices and protocols underscores the need for

new and more effective Intrusion Detection Systems (IDS). Building IoT IDS, on the other hand, necessitates the availability of datasets for processing, training, and evaluating these models.

The first dataset to mimic an MQTT-based network is MQTT-IoT-IDS2020. A simulated MQTT network architecture was used to create the dataset. We are using the MQTT-IoT-IDS2020 dataset in our model. The dataset is available on IEEE Dataport[28]. The dataset contains normal and attacks patterns. In the pre-processing step, we view all the files in the dataset and label the normal with 0 and attack patterns as 1. The rows which contain empty data are deleted in the pre-processing step and then concatenate all the files which we require. Then we create two files, one for training and the other is for testing.

Feature elimination is the next step. A recursive feature elimination algorithm is an algorithm used for feature elimination. Using this algorithm we will eliminate the unwanted features for the model. Using a random forest classifier we choose 13 features and eliminate other features to train our model.

The third step is the creation of zero bias DNN architecture. we have a convolution layer, flatten layer, and dense layer in the architecture. The bias initializers are assigned with value 'zeros' in order to create the zero bias DNN architecture.

The next step is training the system by using the pre-processed data with the help of zero bias DNN.

The final step is testing, the input given to the system is compared with the data used to train the system and produce an output accordingly.

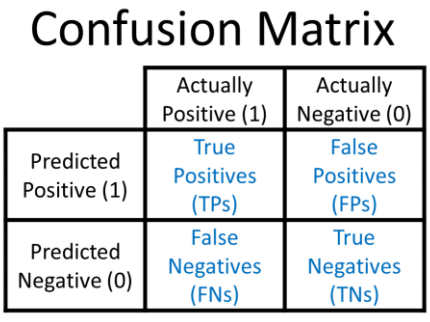
We are using the confusion matrix for the evaluation. A confusion matrix is a method of summarizing a classification algorithm's performance.

If you have an unbalanced amount of observations in each class or if your dataset has more than two classes, classification accuracy alone can be misleading.

Calculating a confusion matrix can help you understand what your classification model is getting right and where it is going wrong.

The number of right and unsuccessful predictions is totaled and broken down by class using count values. The confusion matrix's key is this.

It gives you insight into not only the errors that your classifier is making but also the types of errors that are being made.



A model's accuracy is defined as a statistical measurement scale. The following formula can be used to compute it:
(True positive + True negative/ (True positive + True negative + False positive + False negative)) = Accuracy.
Actual positives represent true values, and the model is correct as well. Negative values are represented by true negative values, and the model will likewise be expected to be negative. The sum of truly negative values is false positive, and the model is correct. The total of genuinely true values is a false negative, and the model is anticipated to be negative.

4 Design Specification

Our proposed plan is to develop an intrusion detection system using the zero bias DNN deep learning method. In a Deep neural network to ensure that the object is appropriately identified, both creative and analytical components of data are studied and categorized. The ML system must alter and derive these components because they are not directly provided to the system.

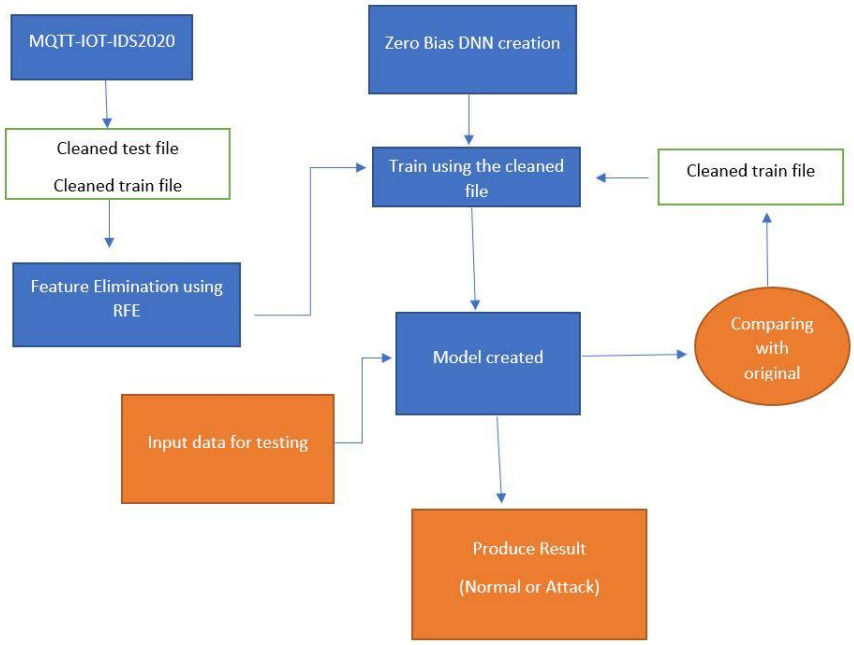


Figure 3

Figure 3 illustrates the design of our model, preprocessing is the first step we have used MQTT-IOT-IDS2020: MQTT INTERNET OF THINGS INTRUSION DETECTION DATASET, to train our intrusion detection system. Communication in IoT devices is commonly done by using Message Queuing Telemetry Transport (MQTT) protocol. Reliable intrusion detection systems are necessary because the number of IoT devices is increasing day by day thus increasing the complexity. The dataset which we have selected contains 5 files. Mqtt_bruteforce.csv, normal.csv, Scan_A.csv, Scan_sU.csv, sparta.csv are the files in

the available dataset. Observing the data, Mqtt_bruteforce.csv and Sparta.csv files contain more accurate data than other files. Many of the columns are having null values in other files. So our system will mainly depend on these two files along with the data of normal cases. All of the files contain 1048576 rows of data each and there are rows which has null values so, in the preprocessing steps, the rows which contain null value are avoided. Data of normal actions are labeled as 0 and the data of attack patterns are labeled as 1. The selected data from the files will be concatenated.

A dataset is typically utilized for more than just training. A single training dataset that has already been processed is frequently divided into numerous portions, which is required to determine how successfully the model was trained. A testing dataset is normally isolated from the data for this reason. A validation dataset can help to avoid training the algorithm on the same sort of data and making biased predictions. Here two files are created, one for training and the other for testing.

Feature elimination-Recursive feature elimination algorithm is an algorithm that helps in eliminating the unnecessary features for the model. The random forest classifier feature of the RFE algorithm is invoked for the elimination purpose.

The Zero Bias DNN creation is one of the important parts of the project. The bias initializer was assigned as zero for the convolutional layer and for one of the dense layers because this assures that all ReLU units fire at the start and therefore obtain and propagate some gradient. If all weights are set to zero, every hidden unit will receive a value of zero, regardless of the input so it will result in producing the wrong answers while initializing zero, only two of the layers were considered.

For the last dense layer, sigmoid function was activated in order to provide the result as 0 or 1.

During training, the model was trained with 80740 samples of data and the smote function has been used to avoid the data imbalance.

In the testing phase, the data entered will be compared with the trained dataset and an output message will be displayed depending on the data either intrusion detected or the normal condition.

5 Implementation

Environmental setup:

Python language is a dynamically semantic object-oriented high-level computer language, is used for coding our model. We have used Anaconda software, a Python distribution that is widely used in data research. Jupyter Notebook was installed using an anaconda navigator to write and execute python codes.

TensorFlow is an open-source software library for machine learning. We have used TensorFlow for the training of our deep neural network.

```
import pandas as pd
import numpy as np
import sys
import sklearn
import io
import random
from sklearn.feature_selection import RFE
from sklearn.ensemble import RandomForestClassifier
import pickle

from sklearn.metrics import accuracy_score,classification_report
from sklearn.metrics import confusion_matrix
from matplotlib import pyplot as plt
import itertools
```

Figure 4

Figure 4 shows the libraries that are imported into our work.

Preprocessing:

The MQTT-IoT-IDS2020 was downloaded from IEEEport and preprocessing was performed. After cleaning the dataset, the combined file was created which contains 40370 normal cases and 30666 attack pattern cases. The file for training contains 80740 samples, and the file for testing contains 17495 samples.

Random forest classifier:

As the name implies, a random forest classifier is a collection of independent decision trees that work together as one. This ensemble tree-based learning approach combines the results of multiple decision trees to estimate the target object's final class. The recursive feature elimination algorithm of the random forest classifier is used in this model.

The features that are present in the datset are the following : 'timestamp', 'src_ip', 'dst_ip', 'protocol', 'ttl', 'ip_len','ip_flag_df', 'ip_flag_mf', 'ip_flag_rb', 'src_port', 'dst_port','tcp_flag_res','tcp_flag_ns','tcp_flag_cwr','tcp_flag_ecn','tcp_flag_urg','tcp_flag_ack','tcp_flag_push','tcp_flag_reset','tcp_flag_syn','tcp_flag_fin','mqtt_messagetype', 'mqtt_messagelength','mqtt_flag_uname','mqtt_flag_passwd','mqtt_flag_retain','mqtt_flag_qo','mqtt_flag_willflag','mqtt_flag_clean', 'mqtt_flag_reserved', 'is_attack' among which 13 features are selected using the RFE algorithm. Figure 5 shows the code for RFE algorithm.

```
from imblearn.over_sampling import SMOTE

oversample = SMOTE()
x_train, y_train= oversample.fit_resample(x_train, y_train)

clf = RandomForestClassifier(n_estimators=20,n_jobs=2)
rfe = RFE(estimator=clf, n_features_to_select=13, step=1)

rfe.fit(x_train, y_train.astype('int'))
X_rfe=rfe.transform(x_train)
true=rfe.support_
rfecolindex=[i for i, x in enumerate(true) if x]
rfecolname=list(colNames[i] for i in rfecolindex)

pickle.dump(rfecolname,open('rfe.pkl','wb'))

x_train=x_train[rfecolname].values
x_test=x_test[rfecolname].values
print(x_test.shape)
print(type(x_test))
```

Figure 5

The architecture of DNN: We have the following for our model:
Conv1D: By convolving the layer input with the convolution kernel over a single spatial (or temporal) dimension, this layer yields a tensor of outputs. When utilize bias is True, a bias vector is created and appended to the outputs. The outputs are active because the activation is 'relu'.
Flatten: It reduces multi-dimensional input tensors to a single dimension, allowing users to model the input layer, build the neural network model, and effectively transmit those inputs to each neuron in the model.
Dense: It's a regular layer of neurons. In our model, we have two dense layers. The sigmoid function is used in the output layer. Figure 6 shows each layer of the model

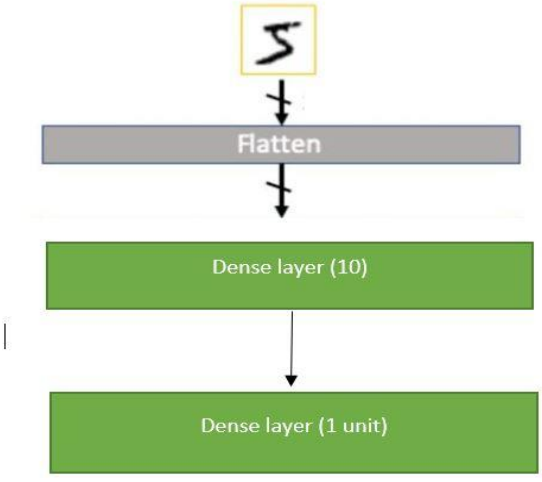


Figure 6

Figure 7 shows the layers used in our Zero bias DNN and its output shape. A total of 114383 parameters are present in the model in which 182 parameters are in the convolutional layer and the rest are on dense layers. The 13 features are used in the model after using RFE are also shown in the figure.

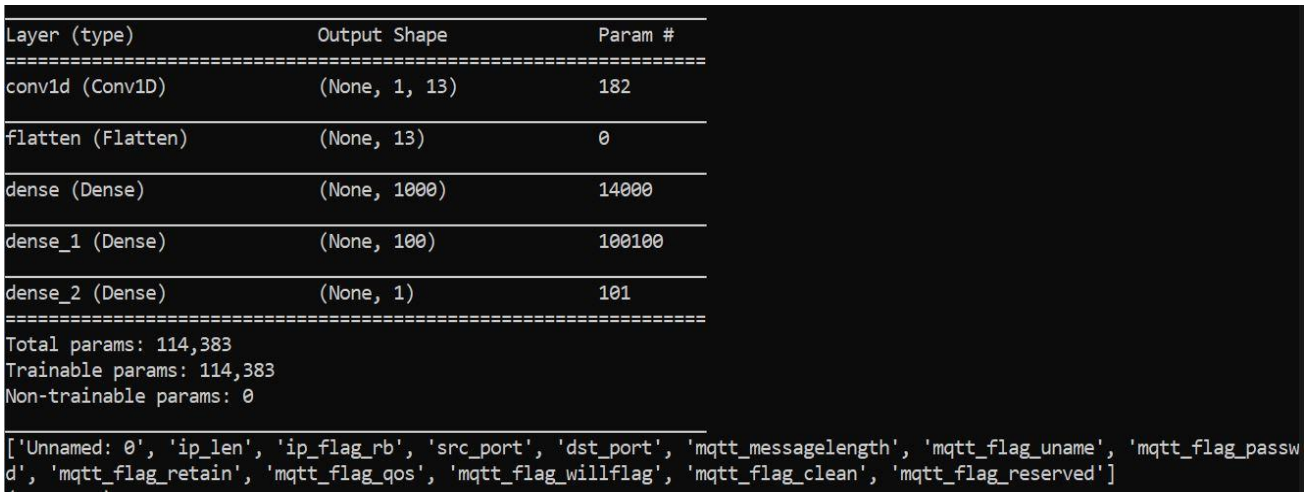


Figure 7

Testing:

When the data is entered into our model a result will be produced showing whether the data is related to features of an attack pattern or not. Figure 8 shows the output screen of our model. In the figure data related to an intrusion was entered and the corresponding output was produced.

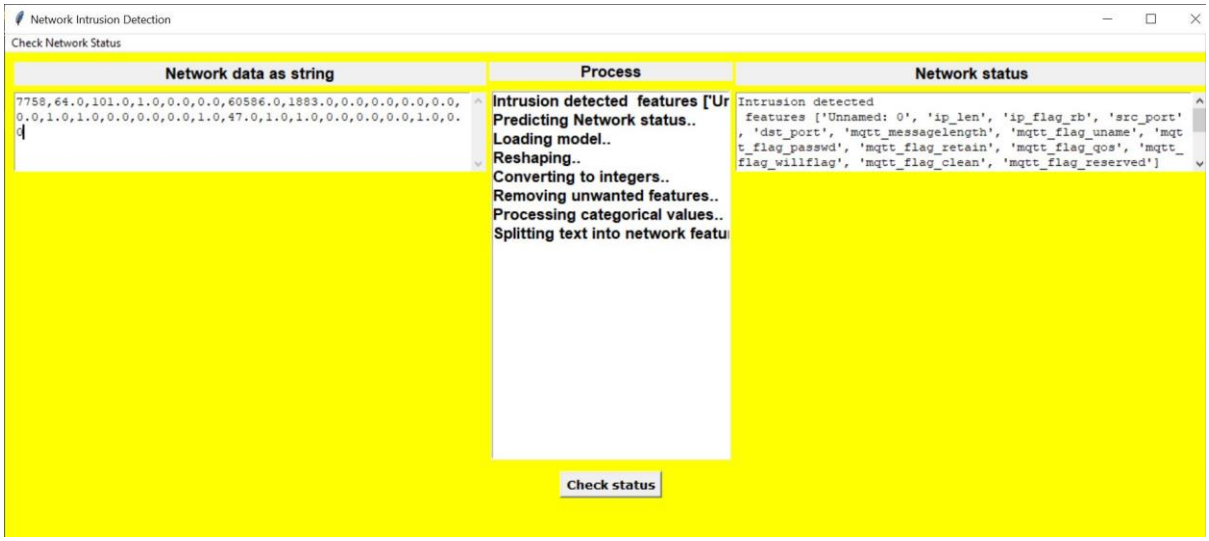


Figure 8

6 Evaluation

The Zero bias DNN for intrusion detection in IoT devices has been implemented and evaluated. The evaluation is done based on the confusion matrix produced. Figure 9 shows the confusion matrix of our model. The aim of performing an experiment is to find the accuracy.

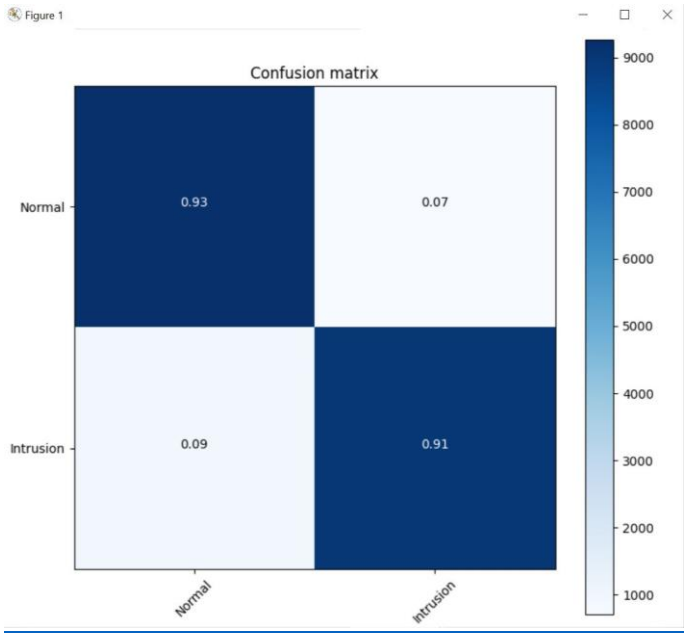


Figure 9

Here in our confusion matrix, we have got two sections. The first row is for normal cases which means the input data will be of features of a normal pattern. The second row is for attack patterns. So according to the confusion matrix, the positive here is normal and the negative is an intrusion.

The value 0.93 represents true positive which means the data entered was of the normal case and the output was also as normal. The 0.07 depicts a false positive where the normal was shown as an attack pattern.

For intrusion, 0.91 is the rate for true negative, which means the intrusion was shown as intrusion itself during the testing. The 0.09 rate shows the false-negative value where the intrusion was shown as a normal pattern.

Accuracy, Recall, Precision, and F1-score can be calculated using the following equations where TP is truly positive, Tn is a true negative, FP is false positive and FN is false negative:

Accuracy = (TP + TN)/(TP + TN + FP + FN)

Recall = TP / (TP + FN)

Precision = TP /(TP + FP)

F1 – score = 2* TP/(2* TP +FP +FN)

TP =0.93, FP=0.07, FN=0.09, TN=0.91
So for our model **accuracy= 0.92, recall=0.911, precision is 0.93 and F1 score is 0.92.**

Research paper	Datasets used	Accuracy	Our Model
Deep Learning Approach for Intrusion Detection System (IDS) in the Internet of Things (IoT) Network using Gated Recurrent Neural Networks (GRU)[9].	KDD Cup '99	93.91	We have used MQTT- IoT IDS 2020 for our Zero Bias DNN model and have an accuracy of 92 %.
A Deep Learning Approach for Intrusion Detection in the Internet of Things using Bi-Directional Long Short-Term Memory Recurrent Neural Network[15].	UNSWNB15	95	
A Novel Intrusion Detection Method for Internet of Things[17].	KDD cup	95%	
Intrusion Detection System for the Internet of Things based on a Machine Learning approach[18].	NSL-KDD cup	Unstable accuracy for each attack type	
MCIDS-Multi Classifier Intrusion Detection system for IoT Cyber Attack using Deep Learning algorithm[19].	UNSW-NB15	88	
Performance Analysis of Deep Learning Algorithms for Intrusion Detection in IoT[20].	NSL-KDD	86	

Table 1

The table shows the accuracy rate and datasets being used in previous papers. None of the papers have used the dataset which we have used. Even though some papers have high accuracy than our paper, the dataset used is too old and contains all the types of attack patterns. Since we concentrate only on IoT devices the MQTT-IoT-IDS2020 is the most

appropriate dataset. The Zero bias DNN model along with the latest dataset, by providing an accuracy of 92% can be considered a good achievement.

7 Conclusion and Future Work

In 2030, the number of Internet of Things (IoT) devices is estimated to reach 80 billion. Smart cities accounted for 28.6 percent of the worldwide IoT market in 2019, followed by industrial purpose at 26.4 percent, health monitoring at 22 percent, smart homes at 15.4 percent, and Internet of Vehicles at 7.7 percent. It is very necessary to have efficient intrusion detection systems to prevent large cyber attacks. A deep neural network provides the best results in intrusion detection. We have developed a new approach zero bias DNN for intrusion detection. Using the latest dataset on IOT- MQTT-IoT-IDS2020, the model was trained and produced output with an accuracy of 92%. This method seems to be a better option for identifying intrusions.
As part of future work, the model can be trained with multiple datasets to produce greater results.

References

[1]C. Liang, B. Shanmugam, S. Azam, M. Jonkman, F. D. Boer, and G. Narayansamy, 'Intrusion Detection System for Internet of Things based on a Machine Learning approach', in *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)*, Mar. 2019, pp. 1–6. doi: 10.1109/ViTECoN.2019.8899448. (accessed Aug. 17, 2021)

[2]S. Manimurugan, S. Al-Mutairi, M. M. Aborokbah, N. Chilamkurti, S. Ganesan, and R. Patan, 'Effective Attack Detection in Internet of Medical Things Smart Environment Using a Deep Belief Neural Network', *IEEE Access*, vol. 8, pp. 77396–77404, 2020, doi: 10.1109/ACCESS.2020.2986013. (accessed Aug. 17, 2021)

[3]G. Parimala and R. Kayalvizhi, 'An Effective Intrusion Detection System for Securing IoT Using Feature Selection and Deep Learning', in *2021 International Conference on Computer Communication and Informatics (ICCCI)*, Jan. 2021, pp. 1–4. doi: 10.1109/ICCCI50826.2021.9402562. (accessed Aug. 17, 2021)

[4]M. R. Babu and K. N. Veena, 'A Survey on Attack Detection Methods For IOT Using Machine Learning And Deep Learning', in *2021 3rd International Conference on Signal Processing and Communication (ICPSC)*, May 2021, pp. 625–630. doi: 10.1109/ICSPC51351.2021.9451740. (accessed Aug. 17, 2021)

[5]'Zero-Bias Deep Learning for Accurate Identification of Internet-of-Things (IoT) Devices | IEEE Journals & Magazine | IEEE Xplore'. https://ieeexplore.ieee.org/document/9173537 (accessed Aug. 17, 2021).

[6]M. Abomhara and G. M. Køien, 'Cyber Security and the Internet of Things: Vulnerabilities, Threats, Intruders and Attacks',

JCSANDM, pp. 65-88-65–88, May 2015, doi: 10.13052/jcsm2245-1439.414. (accessed Aug. 17, 2021)

[7]'Cyber Attacks Classification in IoT-Based-Healthcare Infrastructure | IEEE Conference Publication | IEEE Xplore'. https://ieeexplore.ieee.org/abstract/document/8602974 (accessed Aug. 17, 2021).

[8]G. Gallopeni, B. Rodrigues, M. Franco, and B. Stiller, 'A Practical Analysis on Mirai Botnet Traffic', in
2020 IFIP Networking Conference (Networking), Jun. 2020, pp. 667–668. (accessed Aug. 17, 2021)

[9]M. K. Putchala, 'Deep Learning Approach for Intrusion Detection System (IDS) in the Internet of Things (IoT) Network using Gated Recurrent Neural Networks (GRU)', p. 64. (accessed Aug. 17, 2021).

[10]'Threat analysis of IoT networks using artificial neural network intrusion detection system | IEEE Conference Publication | IEEE Xplore'. https://ieeexplore.ieee.org/abstract/document/7746067 (accessed Aug. 17, 2021).

[11]B. Adhi Tama and K. H. Rhee, 'Attack Classification Analysis of IoT Network via Deep Learning Approach', *Research Briefs on Information & Communication Technology Evolution (ReBICTE)*, vol. 3, Nov. 2017, doi: 10.22667/ReBiCTE.2017.11.15.015. (accessed: Aug 16,2021).

[12]N. Moustafa, B. Turnbull, and K.-K. R. Choo, 'An Ensemble Intrusion Detection Technique Based on Proposed Statistical Flow Features for Protecting Network Traffic of Internet of Things', *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4815– 4830, Jun. 2019, doi: 10.1109/JIOT.2018.2871719. (accessed: Aug 16,2021).

[13]M. AL-Hawawreh, N. Moustafa, and E. Sitnikova, 'Identification of malicious activities in industrial internet of things based on deep learning models', *Journal of Information Security and Applications*, vol. 41, pp. 1–11, Aug. 2018, doi: 10.1016/j.jisa.2018.05.002. (accessed: Aug 16,2021).

[14]A. A. Diro and N. Chilamkurti, 'Distributed attack detection scheme using deep learning approach for Internet of Things',
Future Generation Computer Systems, vol. 82, pp. 761–768, May 2018, doi: 10.1016/j.future.2017.08.043. (accessed: Aug 16,2021).

[15]B. Roy and H. Cheung, 'A Deep Learning Approach for Intrusion Detection in Internet of Things using Bi-Directional Long Short-Term Memory Recurrent Neural Network', in *2018 28th International Telecommunication Networks and Applications Conference (ITNAC)*, Nov. 2018, pp. 1–6. doi: 10.1109/ATNAC.2018.8615294. (accessed: Aug 16,2021).

[16]N. Balakrishnan, A. Rajendran, D. Pelusi, and V. Ponnusamy, 'Deep Belief Network enhanced intrusion detection system to prevent security breach in the Internet of Things', *Internet of Things*, vol. 14, p. 100112, Jun. 2021, doi: 10.1016/j.iot.2019.100112. (accessed: Aug 16,2021).

[17]P. Li and Y. Zhang, 'A Novel Intrusion Detection Method for Internet of Things', in *2019 Chinese Control And Decision Conference (CCDC)*, Jun. 2019, pp. 4761–4765. doi: 10.1109/CCDC.2019.8832753. (accessed: Aug 16,2021).

[18]C. Liang, B. Shanmugam, S. Azam, M. Jonkman, F. D. Boer, and G. Narayansamy, 'Intrusion Detection System for Internet of Things based on a Machine Learning approach', in *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)*, Mar. 2019, pp. 1–6. doi: 10.1109/ViTECoN.2019.8899448. (accessed: Aug 16,2021).

[19] 'MCIDS-Multi Classifier Intrusion Detection system for IoT Cyber Attack using Deep Learning algorithm | IEEE Conference Publication | IEEE Xplore'. https://ieeexplore.ieee.org/document/9388579 (accessed Aug. 17, 2021).

[20] 'Performance Analysis of Deep Learning Algorithms for Intrusion Detection in IoT | IEEE Conference Publication | IEEE Xplore'. https://ieeexplore.ieee.org/document/9484979 (accessed Aug. 17, 2021).

[21]K. Janani and S. Ramamoorthy, 'IoT Security and Privacy Using Deep Learning Model: A Review', in *2021 International Conference on Intelligent Technologies (CONIT)*, Jun. 2021, pp. 1–6. doi: 10.1109/CONIT51480.2021.9498404. (accessed: Aug 16,2021).

[22]I. Ullah and Q. H. Mahmoud, 'Design and Development of a Deep Learning-Based Model for Anomaly Detection in IoT Networks', *IEEE Access*, vol. 9, pp. 103906–103926, 2021, doi: 10.1109/ACCESS.2021.3094024. (accessed: Aug 16,2021).

[23]'Role of Device Identification and Manufacturer Usage Description in IoT Security: A Survey | IEEE Journals & Magazine | IEEE Xplore'. https://ieeexplore.ieee.org/document/9374442 (accessed Aug. 17, 2021).

[24]J. Kim, N. Shin, S. Y. Jo, and S. H. Kim, 'Method of intrusion detection using deep neural network', in *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*, Feb. 2017, pp. 313–316. doi: 10.1109/BIGCOMP.2017.7881684. (accessed: Aug 16,2021).

[25]S. Balakrishna, M. Thirumaran, R. Padmanaban, and V. K. Solanki, 'An efficient incremental clustering based improved K- Medoids for IoT multivariate data cluster analysis', *Peer-to-Peer Netw. Appl.*, vol. 13, no. 4, pp. 1152–1175, Jul. 2020, doi: 10.1007/s12083-019-00852-x.(accessed: Aug 16,2021).

[26] Hindy, H. (2020) 'MQTT-IoT-IDS2020: MQTT Internet of Things Intrusion Detection Dataset'. IEEE. Available at: https://ieee-dataport.org/open-access/mqtt-iot-ids2020-mqtt-internet-things-intrusion-detection-dataset (Accessed: 30 January 2022).