IOT NETWORK SECURITY USING ARTIFICIAL NEURAL NETWORK

ET 7411 PROJECT WORK

PHASE - II

Submitted by

BALAJI K S

Reg. No: 2016212002

A thesis submitted to the

FACULTY OF ELECTRICAL ENGINEERING

in partial fulfilment of the requirements

for the degree of

MASTER OF ENGINEERING

in

EMBEDDED SYSTEM TECHNOLOGIES

Department of Electrical and Electronics Engineering
Faculty of Electrical Engineering
College of Engineering, Guindy,
Anna University, Chennai-600025

MAY 2018

BONAFIDE CERTIFICATE

This is to certify that the thesis titled **"IOT NETWORK SECURITY USING ARTIFICIAL NEURAL NETWORK"** is the bonafide work of **BALAJI K S (Reg. No: 2016212002)** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported here in does not form any part of any other thesis or dissertations on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Dr. G. Uma, Ph.D.,
Professor,
Head of the Department,
Dept. of Electrical and Electronics Engg,
College of Engineering, Guindy,
Anna University, Chennai-600 025.

Dr. R. Ramesh, Ph.D.,
Associate Professor,
Embedded System Technologies,
Dept. of Electrical and Electronics Engg,
College of Engineering, Guindy,
Anna University, Chennai-600 025.

ஆய்வுசுருக்கம்

விசயங்களின் இணையம் என்பது சமீபத்திய போக்கு ஆகும், இது பல்வேறு வகையான கணினி சாதனங்களை உள்ளடக்கிய இணையத்தின் எல்லையை நீட்டிக்கும். 2020 ஆம் ஆண்டளவில் 25 பில்லியன் சாதனங்கள் இணையத்துடன் இணைக்கப்படும் என மதிப்பிடப்பட்டுள்ளது. வலைப்பின்னல் விரைவான வளர்ச்சிக்கான தடையாக பாதுகாப்பின்மை காணப்பட்டுள்ளது. வலைப்பின்னல் பாதுகாப்பு, தகவல் மற்றும் வலைப்பின்னல் தொடர்பான சாதனங்கள் ஆகியவை அடங்கும். நம்பகத்தன்மை மற்றும் ஒருமைப்பாட்டிற்கான வலைப்பின்னல் வழியாக தகவல்களை பாதுகாப்பதற்காக பல குறியாக்க நெறிமுறைகள் பயன்படுத்தப்படுகின்றன. சாதனம் செயலிழப்பு மற்றும் சைபர் தாக்குதல்களை முனைகளில் அல்லது உணரிகள் சாதனங்களில் கண்டறிவதற்கான செயல்திறன் சிறப்பாக வடிவமைக்கப்படவில்லை. இந்த வேலை, ஒரு நரம்பியல் வலைப்பின்னல் மாதிரியை முன்மொழியப்பட்டேன், இது ஐஓடி நெட்வொர்க்கின் விளிம்பு முனையிலிருந்து தவறான தரவை வகைப்படுத்த பயிற்சி அளிக்கப்பட்டது. இது ஒரு அடிப்படையிலான நரம்பியல் வலைப்பின்னல் ஆகும், இது பின்னோக்குப் பின்னான நெறிமுறையில் பயிற்சியளிக்கப்படுகிறது. விளிம்பில் சாதனங்கள் தொடர்ச்சியாக கண்காணிக்கப்படுகின்றன மற்றும் அந்த சாதனத்தின் அசாதாரண மதிப்பு அல்லது கிடைக்கக்கூடிய சாதனத்தை தாக்குபவர் தாக்குதலைத் தாங்கிக்கொள்ளும் சாதனங்களைக் கண்டறிதல் மற்றும் நரம்பு நெட்வொர்க் மாதிரி மூலம் கண்டறியப்பட்ட முனை கண்டறியப்பட்டது.

ABSTRACT

The Internet of Things (IoT) is a recent trend that extends the boundary of the Internet to include a wide variety of computing devices. It is estimated about 25 billion devices will be connected to the internet by 2020. Security has been identified as a potential barrier to the rapid growth of IoT networks. IoT network security involves data and network related devices. Many cryptographic algorithms are used to make data secured through the network for reliability and integrity.

The mechanism to detect device malfunction and cyber-attacks on nodes or sensor devices are not framed effectively. In this work, a neural network model has been proposed which is trained and tested to classify invalid data from edge nodes of an IoT network. It is an MLP based neural network which gets trained on resilient backpropagation algorithm. The edge devices are monitored continually and devices which sense an abnormal value, availability of that device is attacked by an attacker is detected through neural network model.

ACKNOWLEDGEMENT

I express my sincere gratitude to my guide, **Dr. R. Ramesh, Ph.D**., Associate professor, Department of Electrical and Electronics Engineering, Anna University for his guidance, constant encouragement and support. His extensive vision and creative has been a source of inspiration for me throughout this project.

I also thank my project coordinator **Dr. P. Vanaja Ranjan, Ph.D**., Professor, Embedded System Technologies, Department of Electrical and Electronics Engineering, College of Engineering Guindy, Anna University for conducting periodic reviews that helped me in assessing my progress and all the staff of Embedded System Technologies department for providing the most favourable and supportive to carry out the project successfully

I thank **Dr. G. Uma, Ph.D**., Head of Department of Electrical and Electronics Engineering, Anna University for extending all facilities to me to work on this project.

This work would not be complete if I fail to express my indebtedness to almighty, my beloved parents and friends for their moral support and encouragement at all times

BALAJI K S

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ANN	Artificial Neural network
API	Application Peripheral Interface
ARM	Advanced RISC machines
DOS	Denial of Service
GPIO	General Purpose Input/output
HDMI	High definition multi-media interface
IDE	Integrated Development Environment
IOT	Internet of Things
IP	Internet protocol
LAN	Local area network
LED	Light Emitting Diode
MCU	Micro Controller Unit
MLP	Multi-Layer perceptron
MQTT	Message Queuing Telemetry Transport
QOS	Quality of Service
RSSI	Received Signal Strength Indicator
SMTP	Simple Mail Transfer Protocol
SSL	Secure Socket Layers
TLS	Transport Layer Security
TCP	Transmission and control protocol.

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

The Internet of Things (IoT) is a recent trend that extends the boundary of the Internet to include a wide variety of computing devices. This is a network of physical objects embedded with electronics, software, sensors, and connectivity to enable those objects to exchange data with the manufacturer, operator or other connected devices. It is based on the infrastructure of the International Telecommunication Union's Global Standards Initiative (IoT - GSI). It covers devices and objects connected over the Internet Protocol (IP) such as personal computing devices, laptops or desktop computers, tablets, smartphones, and also devices that are connected to each other through non-IP protocols (e.g. Bluetooth, ZigBee). The Internet of Things (IoT) refers to the use of intelligently connected devices and systems to leverage data collected by embedded sensors and actuators in machines and other physical objects over wired and wireless networks.

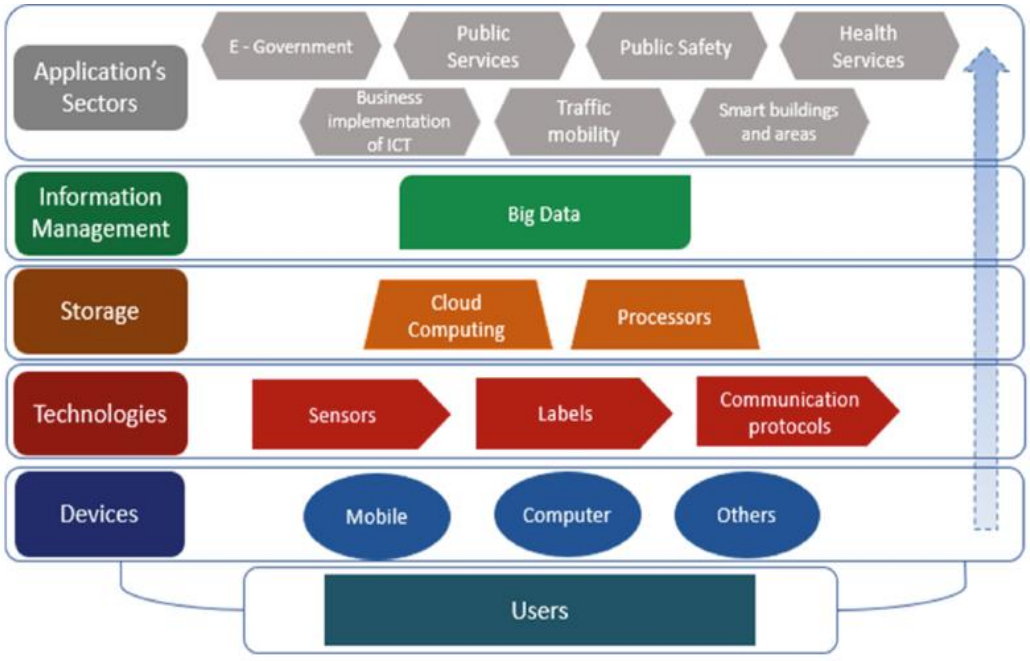

Figure 1.1 Iot Network Architecture

It relays on event-based architecture. Event based is a software architecture pattern promoting the production, detection, consumption and reactions to events. An event can be defined as “as significant change in state”.

1.1.1 SECURITY ISSUES ON IOT NETWORKS

The security flaws of IoT and its ability to perform certain tasks open the door to any associated liability. The three main areas of concern are

- Device malfunction
- Cyber attacks
- Data theft.

These issues can result in a wide variety of damages.

Device Malfunction

IoT introduces a deeper level of automation which can have control over critical systems, and systems impacting life and property. When these systems fail or malfunction, they can cause substantial damage; for example, if an IoT furnace control system experiences a glitch, it may fail in an unoccupied home and cause frozen pipes and water damage. This forces organizations to create measures against it.

Cyber Attacks

IoT devices expose an entire network and anything directly impacted to the risk of attacks. Though those connections deliver powerful integration and productivity, they also create the perfect opportunity for mayhem like a hacked stove or fire safety sprinkler system. The best measures against this address the most vulnerable points, and provide custom protections such as monitoring and access privileges.

Data Theft

Data is IoT networks strength and weakness, proves irresistible to many. These individuals have a number of reasons for their interest: the value of personal data to marketing/advertising, identity theft, framing individuals for crimes, stalking, and a bizarre sense of satisfaction. Measures used to fight attacks are also effective in managing this threat.

Some of the effective measures against attacks are,

- Built-in security
- Encryption
- Authorization
- Risk Analysis

Built-in security, Authorization and encryption schemes are key features which are fixed and varies according to manufacturers of IoT network. But to identify device malfunction we can use a developed firmware to support underlying software.

Risk analysis is the most desirable solution in the context of monitoring edge devices assigned for critical applications. These devices generate data in huge amount and it is difficult to validate using a conventional method.

The main objective is to detect malfunction and attacks on IoT networks by a neural network model. Initially the model is trained before classifying malfunction or any types of attacks.

1.1.2 MACHINE LEARNING

Machine learning usually refers to the changes in systems that perform tasks associated with artificial intelligence (AI). Such tasks involve recognition, diagnosis, planning, robot control, prediction, etc. The changes might be either enhancements to already performing systems or able to synthesis of new systems. As regards

machines, we might say, very broadly, that a machine learns whenever it changes its structure, program, or data (based on its inputs or in response to external information) in such a manner that it's expected future performance improves. Some of these changes, such as the addition of a record to a data base, fall comfortably within the province of other disciplines and are not necessarily better understood for being called learning. But, for example, when the performance of a speech-recognition machine improves after hearing several samples of a person's speech, we feel quite justified in that case to say that the machine has learned.

The methodologies to implement machine learning are,

- Artificial Neural networks
- Genetic algorithms

1.1.3 Artificial Neural Networks

Artificial Neural Network (ANN) is an efficient computing system whose central theme is borrowed from the analogy of biological neural networks. ANNs are also named as "artificial neural systems," or "parallel distributed processing systems". ANN acquires a large collection of units that are interconnected in some pattern to allow communication between the units. These units, also referred to as nodes or neurons, are simple processors which operate in parallel.

Every neuron is connected with other neuron through a connection link. Each connection link is associated with a weight that has information about the input signal. This is the most useful information for neurons to solve a particular problem because the weight usually excites or inhibits the signal that is being communicated. Each neuron has an internal state, which is called an activation signal. Output signals, which are produced after combining the input signals and activation rule, may be sent to other units.

1.1.4 Biological Neuron

A nerve cell (neuron) is a special biological cell that processes information. According to an estimation, there are huge number of neurons, approximately 10^{11} with numerous interconnections, approximately 10^{15} .

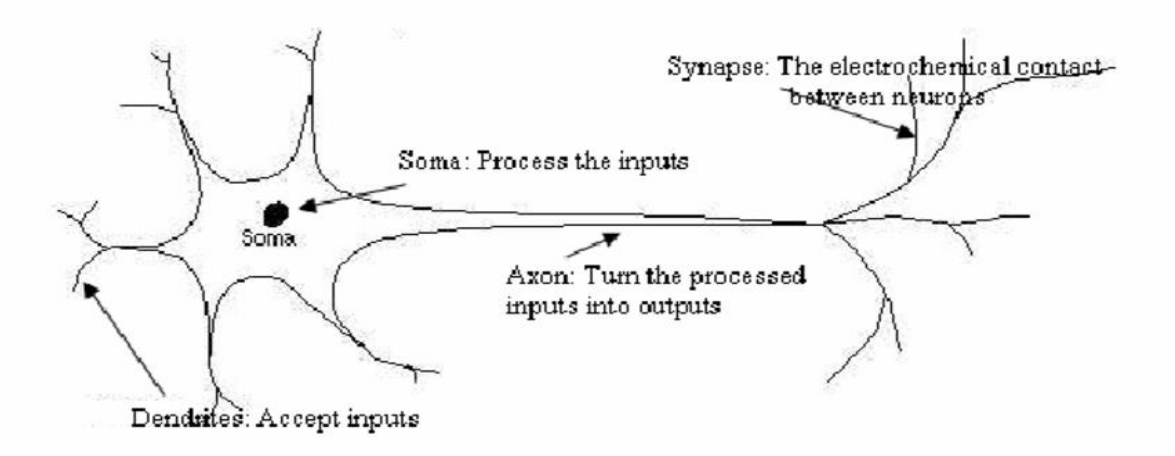

FIGURE 1.2 Schematic Diagram of Biological Neuron

Working of a Biological Neuron

As shown in the above diagram, a typical neuron consists of the following four parts,

Dendrites

They are tree-like branches, responsible for receiving the information from other neurons it is connected to. In other sense, we can say that they are like the ears of neuron.

Soma

It is the cell body of the neuron and is responsible for processing of information, they have received from dendrites.

Axon

It is just like a cable through which neurons send the information.

Synapses

It is the connection between the axon and other neuron dendrites.

The following diagram represents the general model of ANN,

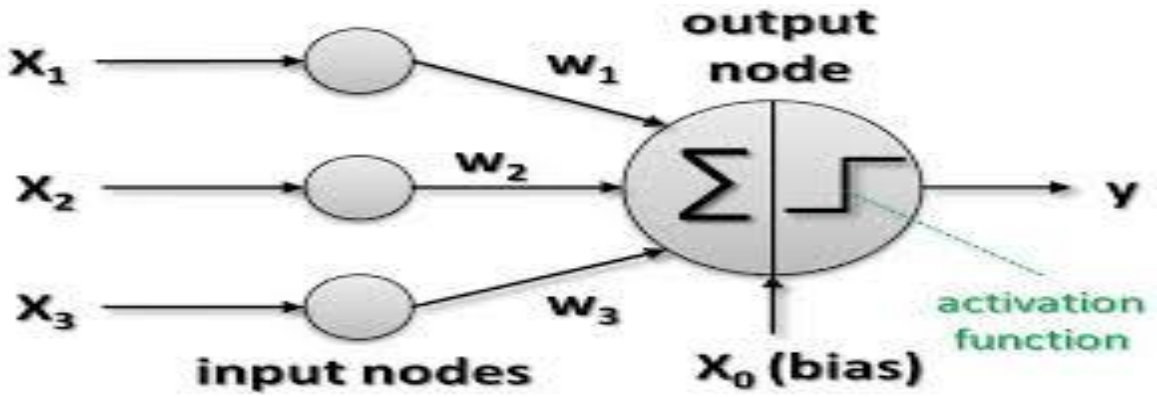

FIGURE 1.3 MODEL OF ANN

Building blocks of ANN

- Network topology
- Activation functions
- Learning

Network Topology

A network topology is the arrangement of a network along with its nodes and connecting lines.

Feed Forward Network

It is a non-recurrent network having processing units/nodes in layers and all the nodes in a layer are connected with the nodes of the previous layers. The connection has different weights upon them. There is no feedback loop means the signal can only flow in one direction, from input to output.

Feedback Network

As the name suggests, a feedback network has feedback paths, which means the signal can flow in both directions using loops. This makes it a non-linear dynamic system, which changes continuously until it reaches a state of equilibrium.

Activation Functions

It may be defined as the extra force or effort applied over the input to obtain an exact output. In ANN, we can also apply activation functions over the input to get the exact output.

Some of the activation functions are,

- Linear
- Sigmoid
- Softmax
- ReLu

Learning

Learning, in artificial neural network, is the method of modifying the weights of connections between the neurons of a specified network

Learning in ANN can be classified into three categories

- Supervised learning
- Unsupervised learning
- Reinforcement learning

1.2 LITERATURE SURVEY

Lulu Liang et al (2016) described the different methods in which Denial of Service (DoS) attack can be implemented on a IoT network. Kali Linux is the attacking tool and different techniques are hping3 with random source IP, simple SYN flood with proofed IP, TCP connect flood.

Prachi Agarwal et al (2013) implemented the artificial neural network using back propagation algorithm to encrypt data over wireless sensor networks.

Mingyuan Xin et al (2015) explained about hybrid cypher algorithm for encryption in an IoT network to provide authenticity and confidentiality of data resources. This algorithm provides high level of security and high speed with less data storage and more suitable for IoT devices.

Janice canedo et al (2016) implemented test bed preparation and use of R programming language as tool to propose a neural network model and detect invalid data points.

Frauke Günther et al (2010) described about novel supervised machine learning algorithms and use of neural net package for training a model. It also discuss about plotting of neural network using ggplot package.

1.3 OBJECTIVE OF THE PROJECT

From above papers it is inferred that many algorithms and methodologies are framed for detection and validation on data, while there is no effective model to detect any anomalies in an IoT network.

- To develop a model which analyse data from a sensor node and validates it.
- To identify device malfunction, cyber-attacks in an IoT network.
- Notify the user with relevant data.

1.4 METHODOLOGIES

- Sense and communicate over MQTT
- Deep learning model
- Detection and display on MQTT broker

1.5 ORGANIZATION OF THESIS

The report consists of six chapters

Chapter 1 presents the overview, literature survey, and objective of the project, methodologies and organization of the thesis.

Chapter 2 describes about the raspberry pi model 3 b and R programming language as a tool to devise the neural network model and about MQTT protocol.

Chapter 4 discuss about the software implementation results of the model and identification of malicious nodes

Chapter 5 discuss about the hardware implementation results of the model and identification of malicious nodes and communicate over MQTT.

Chapter 6 discuss about summary and work done in phase I and II with scope for Future work.

CHAPTER 2

THEORETICAL BACKGROUND AND CONCEPTS

2.1 PROPOSED BLOCK DIAGRAM

As shown in Figure 2.1 an IoT network is designed. It consists of Node MCU as sensor node and Raspberry pi as gateway device. Sensed value are transmitted over MQTT and stored in a CSV file which is feed to gateway device to detect abnormalities. If any malfunction or attack is found then it is intimated to user through mail and data is subscribed by arbitrary MQTT clients.

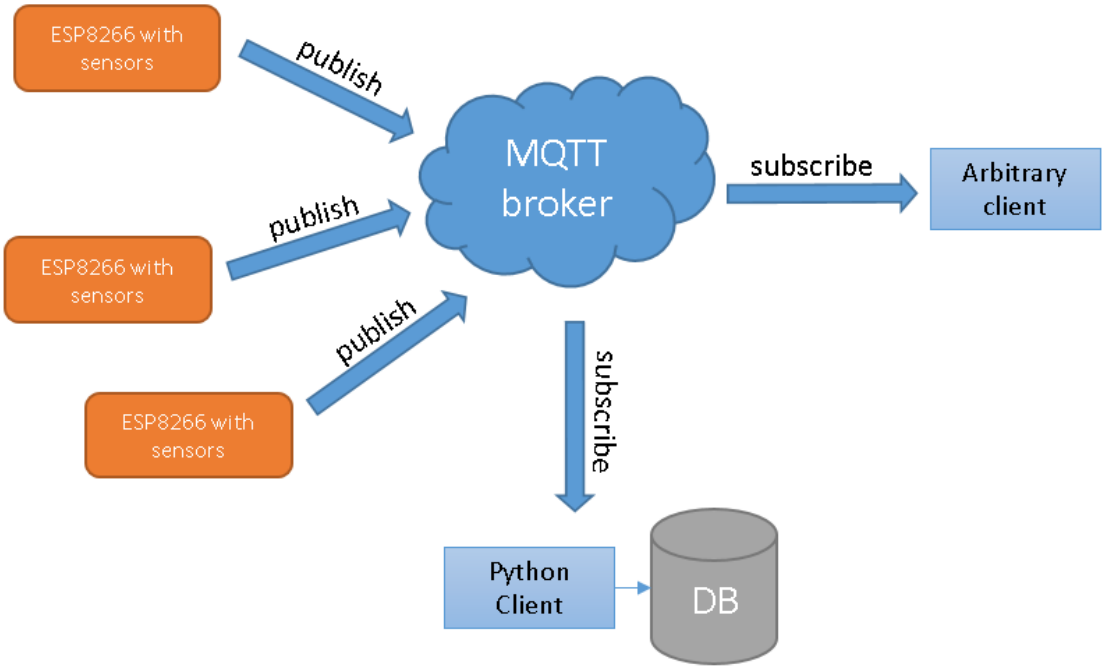

FIGURE 2.1 Proposed Iot Block Diagram

In this project, Raspberry pi model 3B is used which has a built-in Ethernet and WIFI module to connect to internet. Node MCU which has built-in WIFI module is capable of connecting to a network with a unique IP address

of its own. The communication is done through MQTT protocol which is most suited for low bandwidth and energy constrained devices.

The clock speed of 1.2 GHz is required for a neural network code to run effectively. It has to process datasets and also compute thousands of sample at a time so raspberry pi 3 model is chosen.

2.2 RASPBERRY PI

Raspberry pi is a single-board computer. It is a small scale computer in the size little bigger than a credit card, it packs enough power to run games, word processor like open office, image editor like Gimp and any program of similar magnitude. Raspberry Pi was introduced as an educational gadget to be used for prototyping by hobbyists and for those who want to learn more about programming. It certainly cannot be a substitute for our day to day Linux, Mac or Windows PC. It is manufactured and designed in the United Kingdom by the Raspberry Pi foundation with the intention of teaching basic computer science to school students and every other person interested in computer hardware, programming and DIY projects. The Raspberry Pi is manufactured in three board configurations through licensed manufacturing deals with Newark element14 (Premier Farnell), RS Components and Egoman. These companies sell the Raspberry Pi online. Egoman produces a version for distribution solely in China and Taiwan, which can be distinguished from other Pi's by their red colouring and lack of FCC/CE marks. The hardware is the same across all manufacturers.

2.2.1 HARDWARE SPECIFICATION

The Raspberry Pi 3 Model B is the third generation Raspberry Pi. This powerful credit-card sized single board computer can be used for many applications and supersedes the original Raspberry Pi Model B+ and Raspberry Pi 2 Model B. Whilst maintaining the popular board format the Raspberry Pi 3 Model B brings you a more powerful processer, 10x faster than the first generation Raspberry Pi.

Additionally it adds wireless LAN & Bluetooth connectivity making it the ideal solution for powerful connected designs.

The Raspberry Pi 3 model B is based on Broadcom BCM2837(SOC) 64bit ARMv7 Quad Core Processor powered Single Board Computer running at 1.2GHz, 1GB RAM, BCM43143 Wi-Fi on board, Bluetooth Low Energy (BLE) on board, 40pin extended GPIO, 4 x USB 2 ports, 4 pole Stereo output and Composite video port, Full size HDMI, CSI camera port for connecting the Raspberry Pi camera, DSI display port for connecting the Raspberry Pi touch screen display, Micro SD port for loading your operating system and storing data, Upgraded switched Micro USB power source (now supports up to 2.4 Amps), Expected to have the same form factor has the Pi 2 Model B, however the LEDs will change position.

FIGURE 2.2 RASPBERRY PI 3 MODEL B

2.2.2 COMPATIBLE OPERATING SYSTEMS

Some of the operating systems suitable for Raspberry Pi are discussed below.

Raspbian

Raspbian is a free operating system based on Debian optimized for the Raspberry Pi hardware. An operating system is the set of basic programs and utilities that make your Raspberry Pi run. However, Raspbian provides more than a pure OS: it comes with over 35,000 packages, pre-compiled software bundled in a nice format for easy installation on your Raspberry Pi. This project runs on latest version of raspbian.

Pidora

Pidora is another Linux distribution like Raspbian, but is based on the Fedora distribution. It gives you a different look and feel to Raspbian. The current build is for the ARMv6 architecture, and therefore will not run on the Pi2.

Risc OS

Risc OS operating system is different from the others in the fact that it is not based on Linux, but is instead a completely separate OS. It was originally designed by Acorn in Cambridge and has links to the team that developed the original ARM microprocessors

Snappy Ubuntu core

The advent of the ARMv7 in the Raspberry Pi2, a version of the Ubuntu Linux operating system has become available. This is an early, alpha release, which means that it is not really intended for everyday users, but more for developers to start developing "snappy" apps for Ubuntu.

Arch Linux ARM

Arch Linux is another distribution for more experienced users. The base OS is minimal and needs additional packages to be installed by the user to make up the OS in to a full environment. However it has the reputation of being a good stable distribution.It is specially meant for Raspberry Pi 3. Raspbian has been updated to the new stable version of Debian, which is called Jessie. Boots straight to desktop instead of the command line. Can be changed back using the Raspberry Pi Configuration application.

2.3 NODE MCU

NodeMCU was created shortly after the release of ESP8266 WIFI module. The ESP8266 module is a Wi-Fi SoC integrated with a Tensilica Xtensa (LX106 core), widely used in IoT applications. Node MCU is an open source hardware with ESP-12 core with 4MB of flash.

FIGURE 2.3 NODEMCU with WIFI Module

2.3.1 SPECIFICATIONS OF NODEMCU

- NodeMCU supports 802.11 b/g/n WiFi standards
- It has on-chip board antenna and RF bauln
- It has 10 digit ADC and supports IPV4, HTTP, UDP, TCP and FTP network protocols.
- Operating voltage is (3.3 – 5) V and operating frequency is (2.4 – 2.6 GHz).
- On board power management modules, PLL, regulator, Power amplifier, Noise filters which makes it a less external dependent circuitry interface.

Some of the applications of Node MCU are Home appliances control, used in IP cameras and wearable electronics, security tags , Wi-Fi location and position aware devices.

2.4 DIGITAL TEMPERATURE SENSOR

The DS18B20 digital thermometer provides 9-bit to 12-bit Celsius temperature measurements and has an alarm function with non-volatile user-programmable upper and lower trigger points. It is a water, dust proof which can be used in any weather conditions. It has data , supply and ground pin. A resistor is mandatory between data and supply pin to read temperature value and sensor is shown in figure 2.4.

Fig 2.4 Temperature Sensor DS18B20

2.5 SOFTWARE

The software tool used is R, a statistical and machine learning tool. It has lot of packages to support machine learning. Packages used are,

- Neuralnet
- Keras
- mailR

It also supports machine learning algorithms like back-propagation and optimizers to minimize error. mailR package is used to send notification from R to user defined mail-ID.

2.5.1 PROGRAMMING IN R

R is an open-source programming language and software environment for statistical computing and graphics that is supported by the R Foundation for Statistical Computing. The R language is widely used among statisticians and data scientists for developing statistical software and data analysis.

2.5.2 FEATURES OF R

R has an effective data handling and storage facility. It suite's operators for calculations on arrays, in particular matrices. A large, coherent, integrated collection of intermediate tools for data analysis, graphical facilities for data analysis and display either directly at the computer or on hardcopy,

A well developed, simple and effective programming language (called 'S') which includes conditionals, loops, user defined recursive functions and input and output facilities. (Indeed most of the system supplied functions are themselves written in the S language.)

2.5.3 R STUDIO

The R Studio project currently provides most of the desired features for an IDE in a novel way, making it easier and more productive to use R. The R Studio program can be run on the desktop or through a web browser. The desktop version is available for Windows, Mac OS X, and Linux platforms and behaves similarly across all platforms, with minor differences for keyboard shortcuts.

2.5.4 WORKING WITH R STUDIO

The source-code editor is feature-rich and integrated with the built-in console. The main components of an IDE are all nicely integrated into a four-panel layout that includes a console for interactive R sessions, a tabbed source-code editor to organize a project's files, and panels with notebooks to organize less central components The console and source-code editor are tightly linked to R's internal help system through

tab completion and the help page viewer component. RStudio provides many convenient and easy-to-use administrative tools for managing packages, the workspace, files, and more.

RStudio is much easier to learn than Emacs/ESS, easier to configure and install than Eclipse/StatET, has a much better editor than JGR, is better organized than Sciviews, and unlike Notepad++ and RGui, is available on more platforms than just Windows.

2.5.5 PACKAGES IN R

All R functions and datasets are stored in packages. Only when a package is loaded are its contents available. This is done both for efficiency (the full list would take more memory and would take longer to search than a subset), and to aid package developers, who are protected from name clashes with other code. The standard (or base) packages are considered part of the R source code. They contain the basic functions that allow R to work, and the datasets and standard statistical and graphical functions available in R. There are thousands of contributed packages for R, written by many different authors. Some of these packages implement specialized statistical methods, others give access to data or hardware, and others are designed to complement textbooks.

2.5.5.1 NEURALNET PACKAGE

The package neural net contains a very flexible function to train feed forward neural networks, i.e. to approximate a functional relationship in the above situation. It can theoretically handle an arbitrary number of covariates and response variables as well as of hidden layers and hidden neurons even though the computational costs can increase exponentially with higher order of complexity. This can cause an early stop of the iteration process since the maximum of iteration steps, which can be defined by the user, is reached before the algorithm converges.

The package neural net focuses on multi-layer perceptron, which are well applicable when modelling functional relationships. The underlying structure of an MLP is a directed graph, i.e. it consists of vertices and directed edges, in this context called neurons and synapses. The neurons are organized in layers, which are usually fully connected by synapses. In neural net, a synapse can only connect to subsequent layers. The input layer consists of all covariates in separate neurons and the output layer consists of the response variables. The layers in between are referred to as hidden layers, as they are not directly observable. Input layer and hidden layers include a constant neuron relating to intercept synapses, i.e. synapses that are not directly influenced by any covariate.

Neural networks are fitted to the data by learning algorithms during a training process. It focuses on supervised learning algorithms. These learning algorithms are characterized by the usage of a given output that is compared to the predicted output and by the adaptation of all parameters according to this comparison.

A widely used learning algorithm is the resilient back propagation algorithm. The resilient back propagation algorithm is based on the traditional back propagation algorithm that modifies the weights of a neural network in order to find a local minimum of the error function. Therefore, the gradient of the error function (dE/dw) is calculated with respect to the weights in order to find a root. In particular, the weights are modified going in the opposite direction of the partial derivatives until a local minimum is reached.

2.5.5.2 KERAS PACKAGE

Keras is a high-level neural networks API, developed with a focus on enabling fast experimentation. It allows easy and fast prototyping. It offers consistent & simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear and actionable feedback upon user error.

A model is understood as a sequence or a graph of standalone, fully-configurable modules that can be plugged together with as little restrictions as

possible. In particular, neural layers, cost functions, optimizers, initialization schemes, activation functions, regularization schemes are all standalone modules that you can combine to create new models.

2.5.5.3 MAILR PACKAGE

The package mailR is a utility to send mails from R. It interfaces with Apache commons software to send e-mails. Send.mail object has many attributes like from, to, subject, body, encoding, smtp, authenticate and attach.files.

The only mandatory value in the list 'smtp' is host.name that is the SMTP server address. A port number can also be provided via the list item 'port'. In case the SMTP server requires authorization, the parameter 'authenticate' must be set to TRUE and the list 'smtp' must include items 'user.name' and 'passwd'.

If SSL or TLS encryption is required by the SMTP server, these can be indicated by setting a list item 'ssl' as TRUE or 'tls' as TRUE respectively. Attach.files is used to add files and additionally the description can also be provided. At last send attribute is set TRUE to send mail.

2.6 MQTT PROTOCOL

MQTT stands for MQ Telemetry Transport. It is a publish/subscribe, extremely simple and lightweight messaging protocol, designed for constrained devices and low-bandwidth, high-latency or unreliable networks. MQTT is a Client Server publish/subscribe messaging transport protocol. It is light weight, open, simple, and designed so as to be easy to implement. These characteristics make it ideal for use in many situations, including constrained environments such as for communication in Machine to Machine (M2M) and Internet of Things (IoT) contexts where a small code footprint is required and/or network bandwidth is at a premium. The design principles are to minimise network bandwidth and device resource requirements whilst also attempting to ensure reliability and some degree of assurance of delivery.

These principles also turn out to make the protocol ideal of the emerging “machine-to-machine” (M2M) or “Internet of Things” world of connected devices, and for mobile applications where bandwidth and battery power are at a premium.

MQTT has been widely implemented across a variety of industries since 1999. The protocol runs over TCP/IP, or over other network protocols that provide ordered, lossless, bi-directional connections. Its features include use of the publish/subscribe message pattern which provides one-to-many message distribution and decoupling of applications A messaging transport that is agnostic to the content of the payload. A small transport overhead and protocol exchanges minimized to reduce network traffic. A mechanism to notify interested parties when an abnormal disconnection occurs.

There are three types of QoS provided by MQTT as shown below,

- "At most once", where messages are delivered according to the best efforts of the operating environment. Message loss can occur. This level could be used, for example, with ambient sensor data where it does not matter if an individual reading is lost as the next one will be published soon after.
- "At least once", where messages are assured to arrive but duplicates can occur.
- "Exactly once", where message are assured to arrive exactly once. This level could be used, for example, with billing systems where duplicate or lost messages could lead to incorrect charges being applied.

CHAPTER 3

SOFTWARE IMPLEMENTATION OF DETECTING DEVICE MALFUNCTION

In this work, device malfunction and cyber-attacks in IoT network is identified by means of validating data using statistical R programming tool.

3.1 FLOWCHART

The flowchart of proposed system is showed in figure 3.1

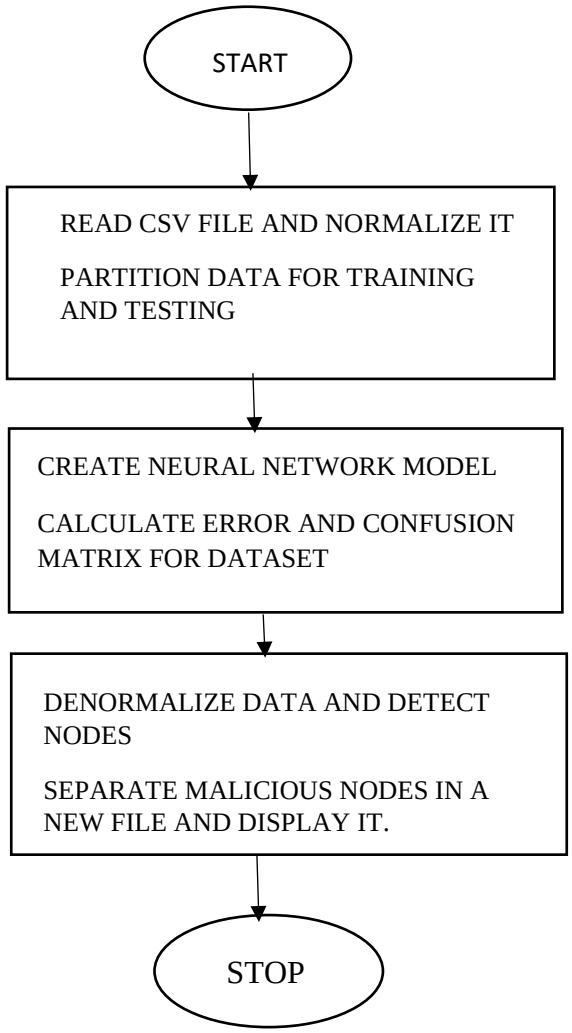

FIGURE 3.1 Flowchart for Implementing Malfunction

3.2 NORMALIZATION OF DATA

The neural network model accepts input in range of 0 to 1. So we normalize RSSI and temperature of inlet and outlet of valve using a normalization function. The normalized function is then feed to an activation function to activate the neuron. The normalized histogram of inlet temperature is shown in Figure 3.2.

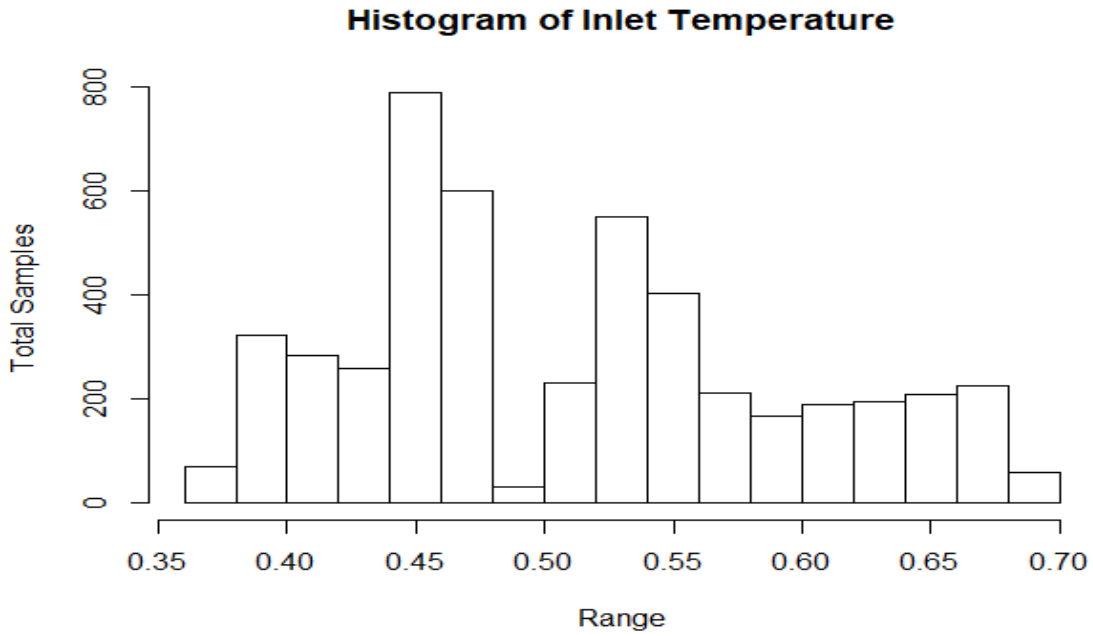

FIGURE 3.2 Histogram of Inlet Temperature

The temperature from out valve has to be normalized in order to feed it as an input to neural network. Normalized histogram of outlet temperature is shown in Figure 3.3.

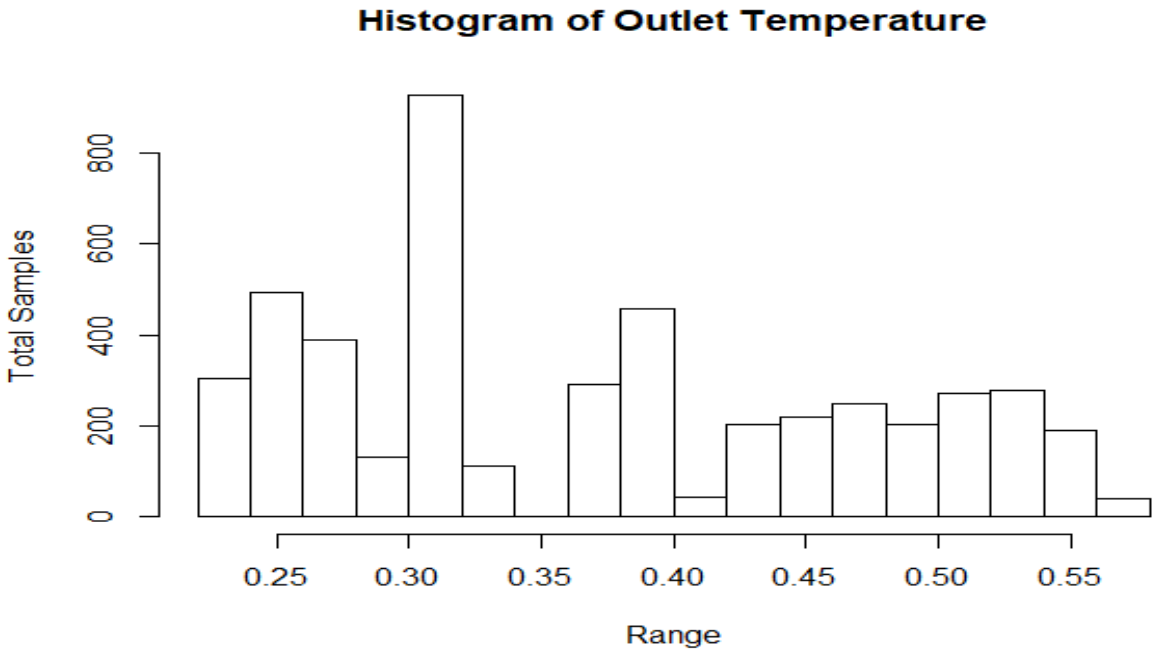

FIGURE 3.3 Histogram of Outlet Temperature

RSSI is the signal strength of Nodemcu with broker in dBm and comprised of negative values. The normalized histogram of signal strength is shown in figure 3.4

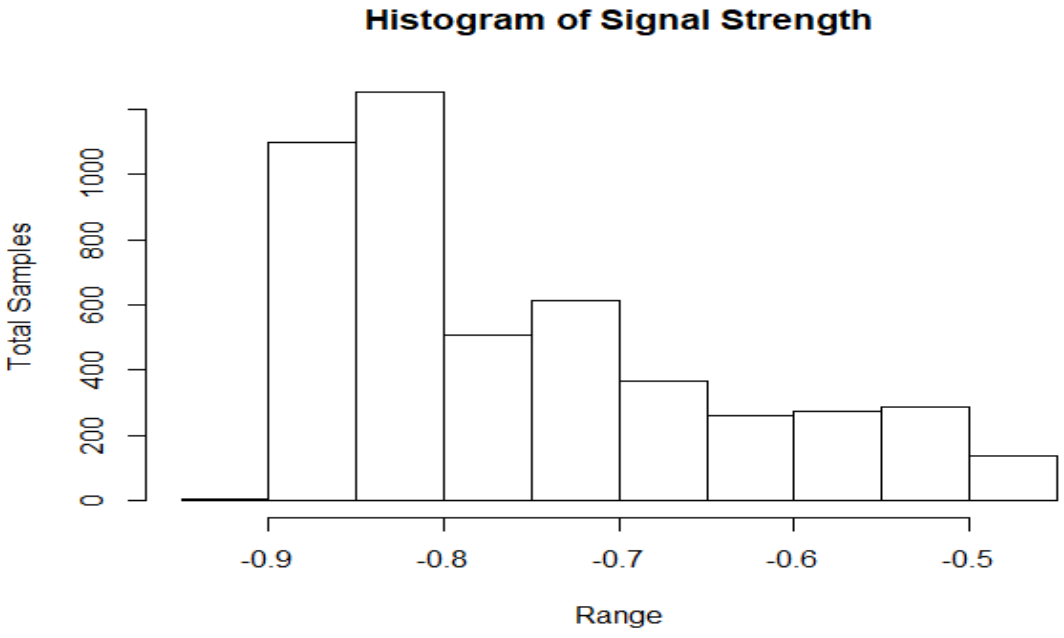

FIGURE 3.4 Histogram of RSSI Signal Strength

3.3 NEURAL NETWORK MODEL

The phase I uses the neural-net package to create a neural network model to classify malfunction using device ID and temperature as input parameters.

This neural network model shown in figure 3.5 has two hidden layers which comprise of 7×2 neurons. Resilient back propagation algorithm is used to minimize error and weights are updated for each iteration. Based on result devices are categorized into valid and in-valid temperature values. Once this is done a data frame of device with their corresponding temperature values is written to a new excel file. A package called mailR is used to send email of the collected data to the registered email.

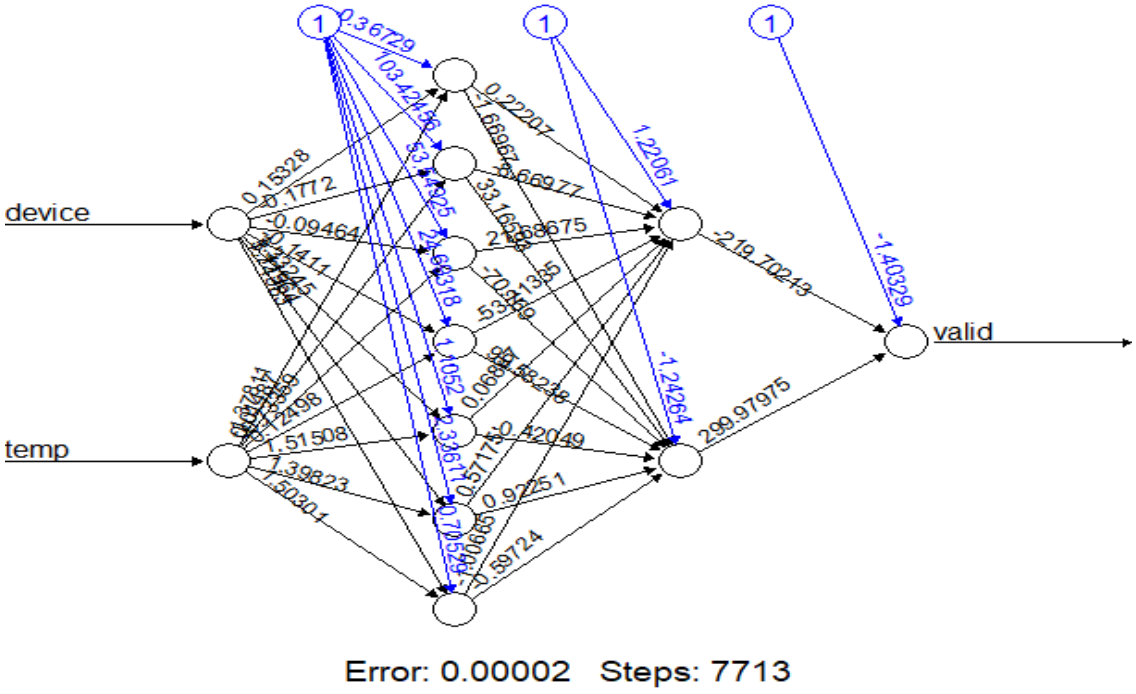

FIGURE 3.5 Modified Neural Network Model

The temperature sensor value is not sufficient to detect cyber attacks in an iot network. Neural-net package has no built in normalization technique for negative values and its difficult to normalize with user defined function,so keras package which is a API based machine learning package is used and backend computations are implemented using tensorflow sponsered by google.

In phase II a keras based sequential model with temperatures and RSSI value as inputs is feed into the neural network model. It is trained with sample data and tested for real data from edge devices. Optimization techniques are formulated inorder to increase the efficiency of the model.

A deep learning model with six layers and 113 neurons is designed using keras. The validation split is done with 20 % of training data and epoch or repition is set high. Adam optimizer and accuracy is set as a metric to evaluate model performance. The plot of evaluated model with loss and accuracy as parameter for validation split is shown in figure 3.6

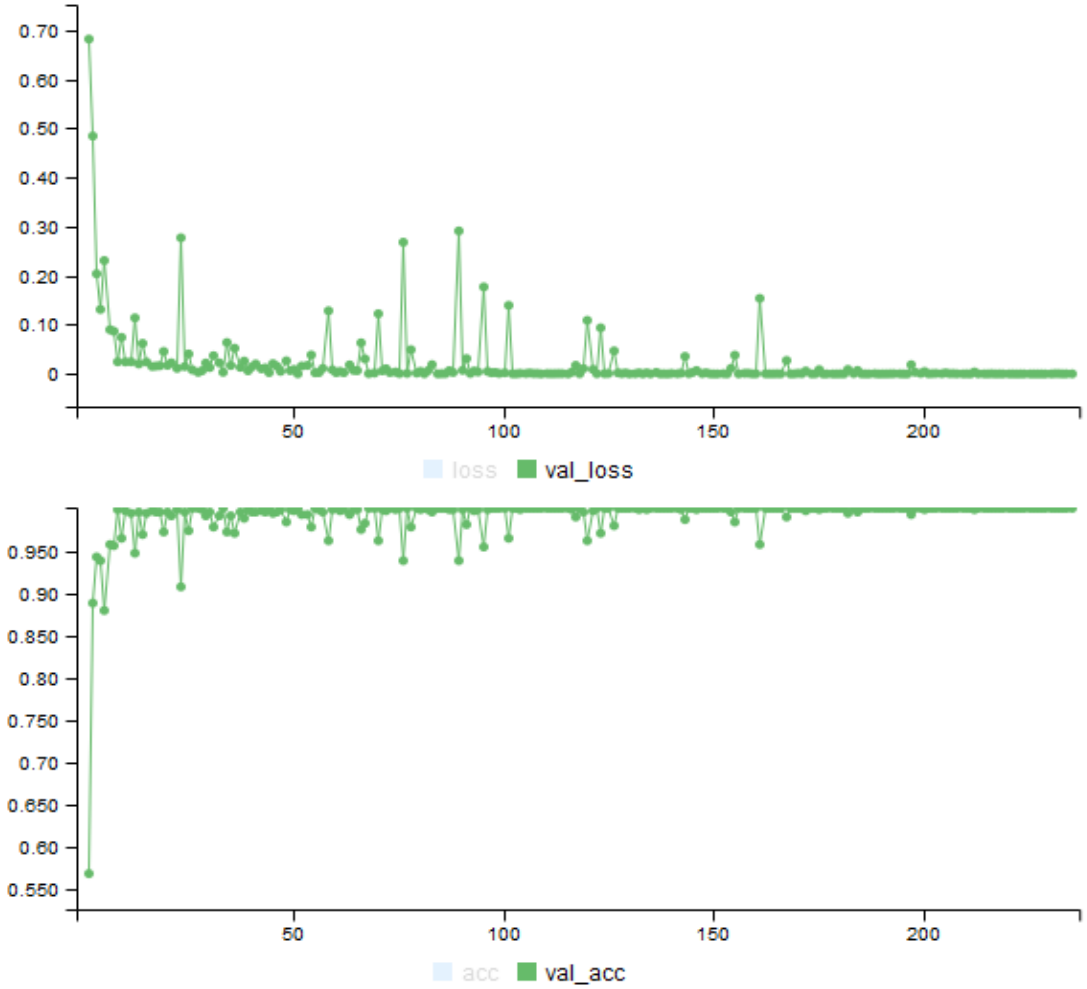

FIGURE 3.6 Plot of Evaluated Model with Respect To Validation Split

The plot of evaluated model with loss and accuracy as parameter for entire samples is shown in figure 3.7. It is seen that intially the loss is high, as the number of epoches(iteration) increases the losses get reduced exponentially and remained constant till the end of epoch cycle.

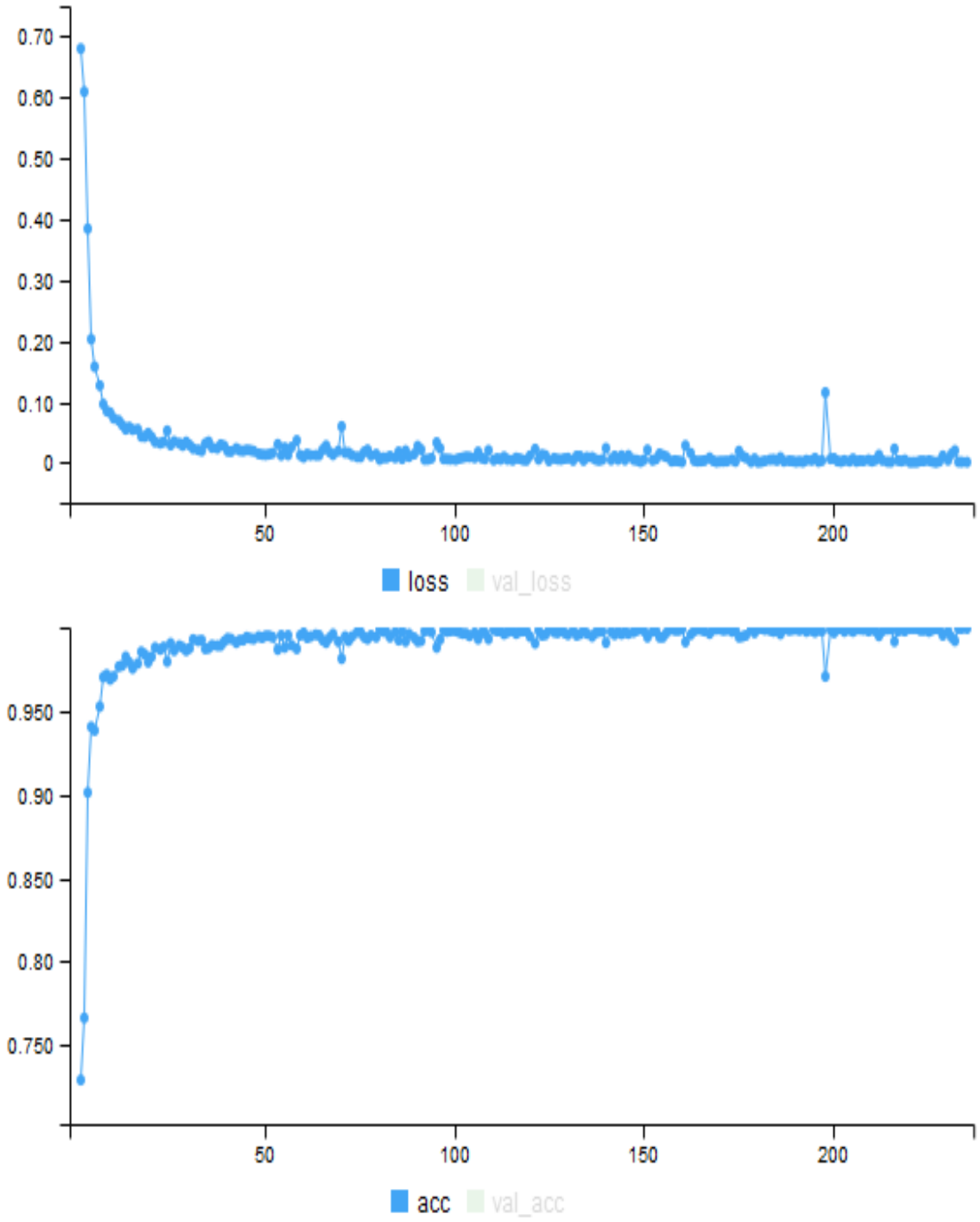

FIGURE 3.7 Plot of Evaluated Model with Respect to All Samples

There are five optimization techniques available but adam is chosen because of its robust optimization and depends on exponentially decaying average of past gradients with adaptive learning rate for each parameter.The mathematical equation for adam optimizer is shown below,

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \quad (1)$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)\, g_t^2 \quad (2)$$

Where m_t is the estimates of first moment (mean) and v_t is the estimates of second moment (un-centered variance) of the gradients.

CHAPTER 4

HARDWARE IMPLEMENTATION OF IOT NETWORK AND MALFUNCTION

4.1 NODEMCU INTERFACING WITH DS18B20

DS18B20 is a water resistant temperature sensor which has three channels such as power, ground and data channel. Digital pin of node MCU is connected to data pin of the sensor and a 4.7K resistor between power and data is necessary to read temperature value using this sensor. Connection setup is shown in fig 4.1 below

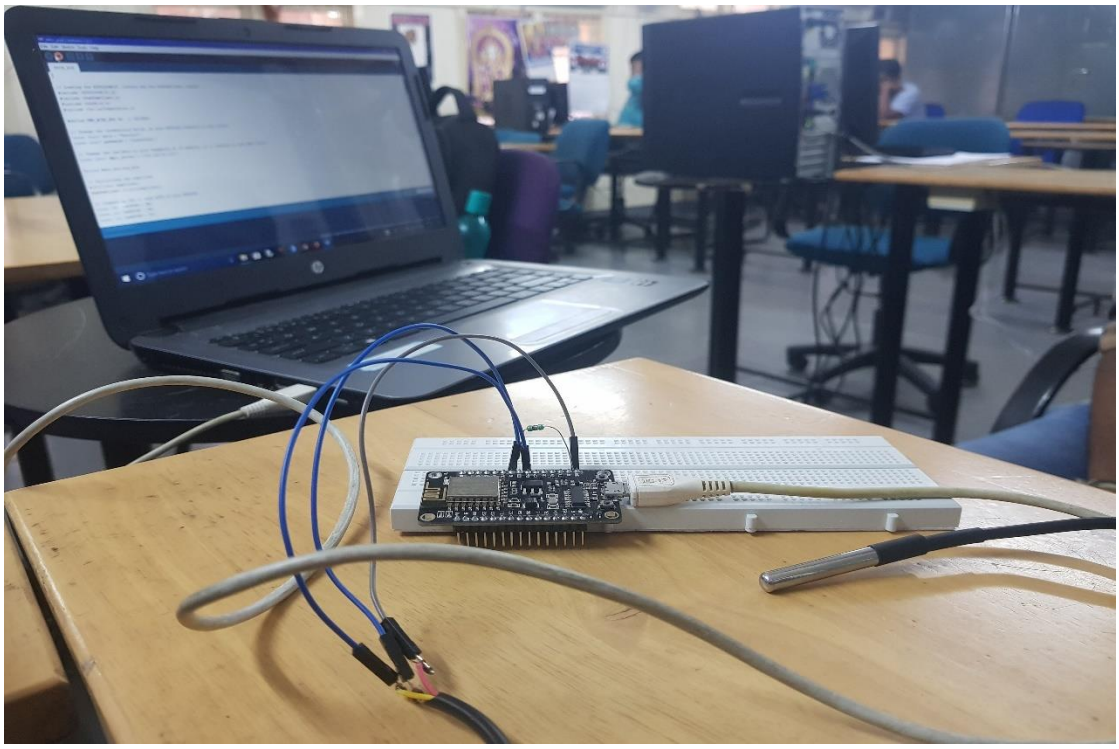

FIGURE 4.1 SETUP OF NODEMCU WITH DS18B20

The software used is Arduino IDE, the board package for node mcu is installed and necessary packages are included in the script. The temperature value is sensed and displayed on serial monitor is shown in figure 4.2

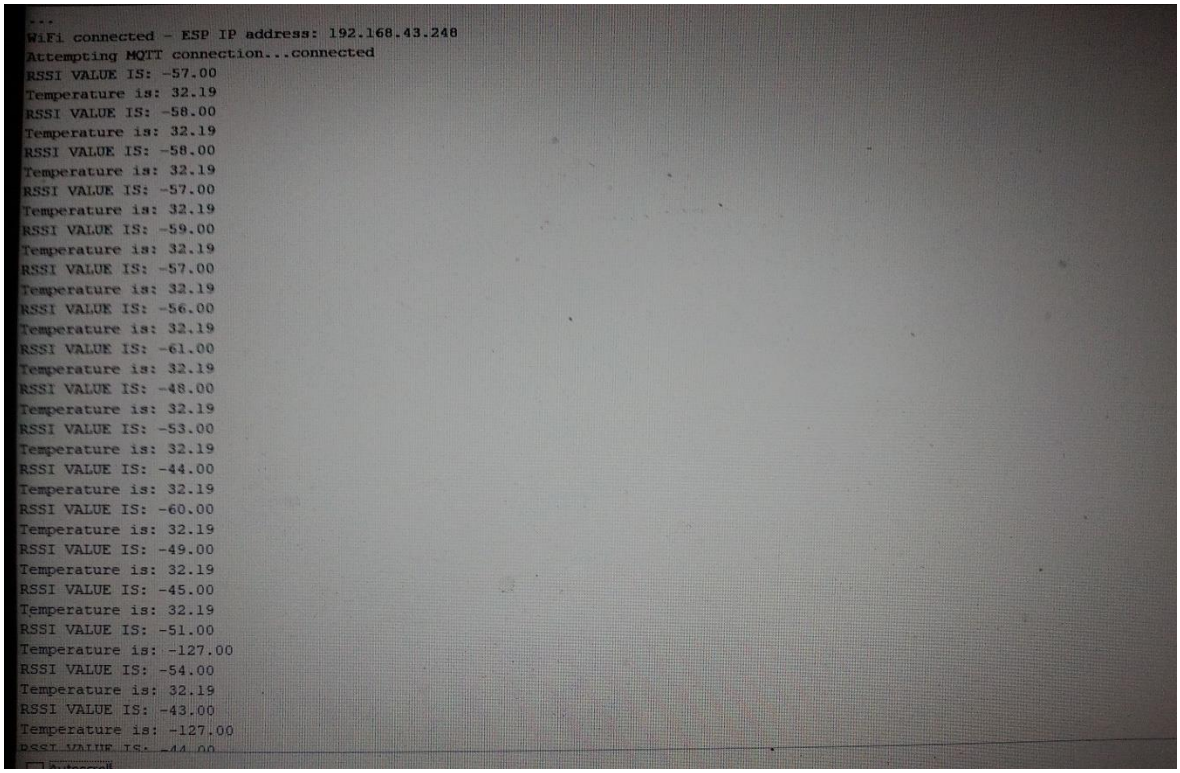

FIGURE 4.2 DS18B20 SENSOR VALUE

4.2 DEMONSTRATING MQTT THROUGH WEBSERVER

In a Iot network, gateway and sensor nodes communicate through MQTT protocol which runs on top of TCP/IP. To demonstrate how MQTT works a python based client (web-server) subscribe and publish messages to valid topics.

The LED's are connected to digital pins of node mcu and are controlled from web-server to turn it ON/OFF. If the message published is 1 then LED is turned ON else if it is 0 then LED is turned OFF. This is shown in figure 4.3

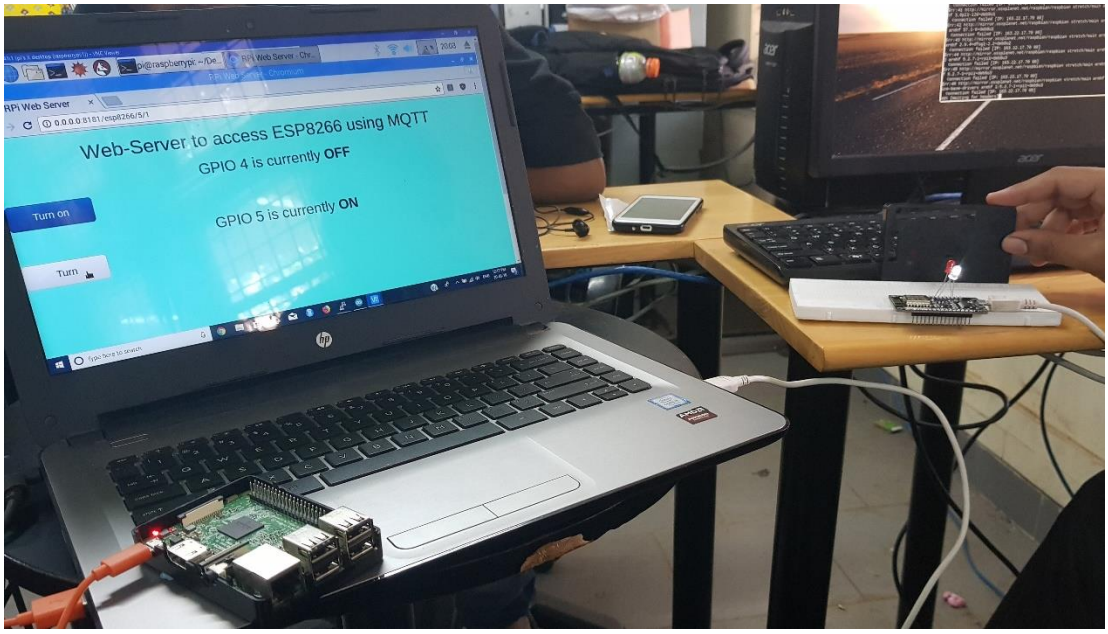

FIGURE 4.3 MQTT BASED WEBSERVER

4.3 MQTT BROKER SUBSCRIBE TO NODE MCU

The temperature and RSSI value is sent to raspberry pi (MQTT broker) with the help of PubSub library. MQTT broker can log those values by subscribing to valid topics.A sample subscribtion of those values is shown in figure 4.4

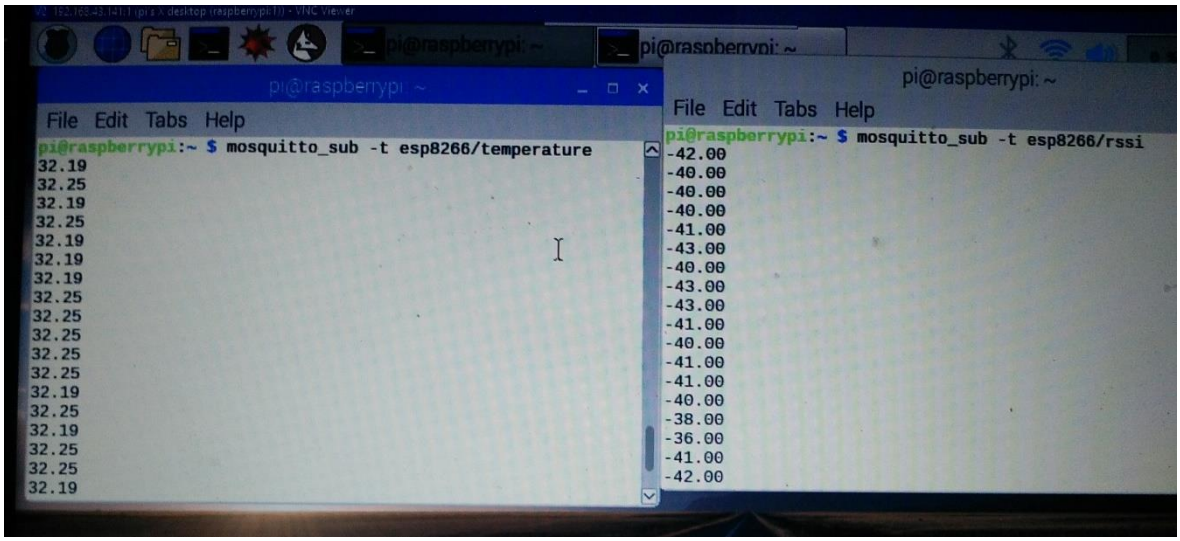

FIGURE 4.4 SUBSCRIBTION TO SENSED VALUES OVER MQTT

CHAPTER 5

RESULTS AND DISCUSSIONS

The Internet of Things (IoT) is a recent trend that extends the boundary of the Internet to include a wide variety of computing devices. Security has been identified as a potential barrier to the rapid growth of IoT networks. The edge devices are monitored continually and devices which sense an abnormal value, availability of that device is attacked by an attacker is detected through neural network model.

5.1 SOFTWARE IMPLEMENTATION FOR DEVICE MALFUNCTION

A Deep learning model using keras package is proposed to implement malfunction and cyber-attacks in a network. The accuracy of the model is improved using optimization techniques. Model with accuracy of 1 is achieved after repetitive optimization of layers and epochs. The plot of final model with greater accuracy is shown inn figure 5.1

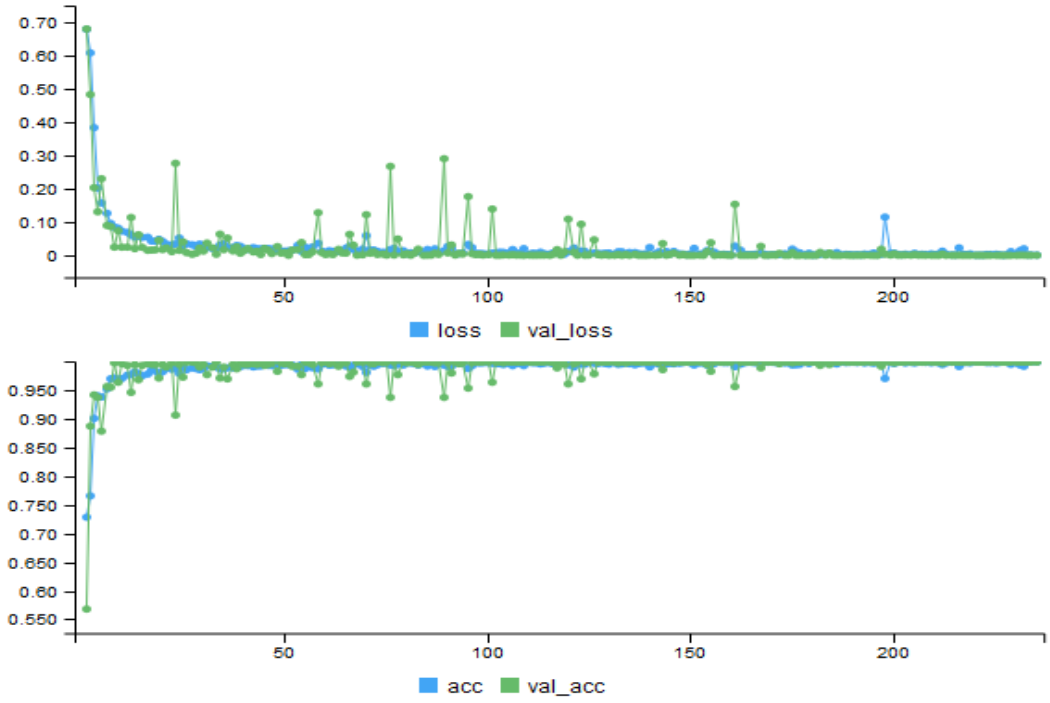

FIGURE 5.1 PLOT OF KERAS MODEL

5.2 HARDWARE IMPLEMENTATION OF DEVICE MALFUNCTION

The temperature sensor value from inlet , outlet and RSSI of the valve is continiously measured and logged into a csv file of pi. The data are transmitted over mqtt and gateway device feeds these values to the loaded neural network model.Once validation of data is done, through MQTT the abnormal values and attack on devices based on RSSI is obtained and published to broker over mqtt. Any client subscribed to those topics can fetch those data and control its operation. The abnormal values which are subscribed to their respective topics is shown in figure 5.2

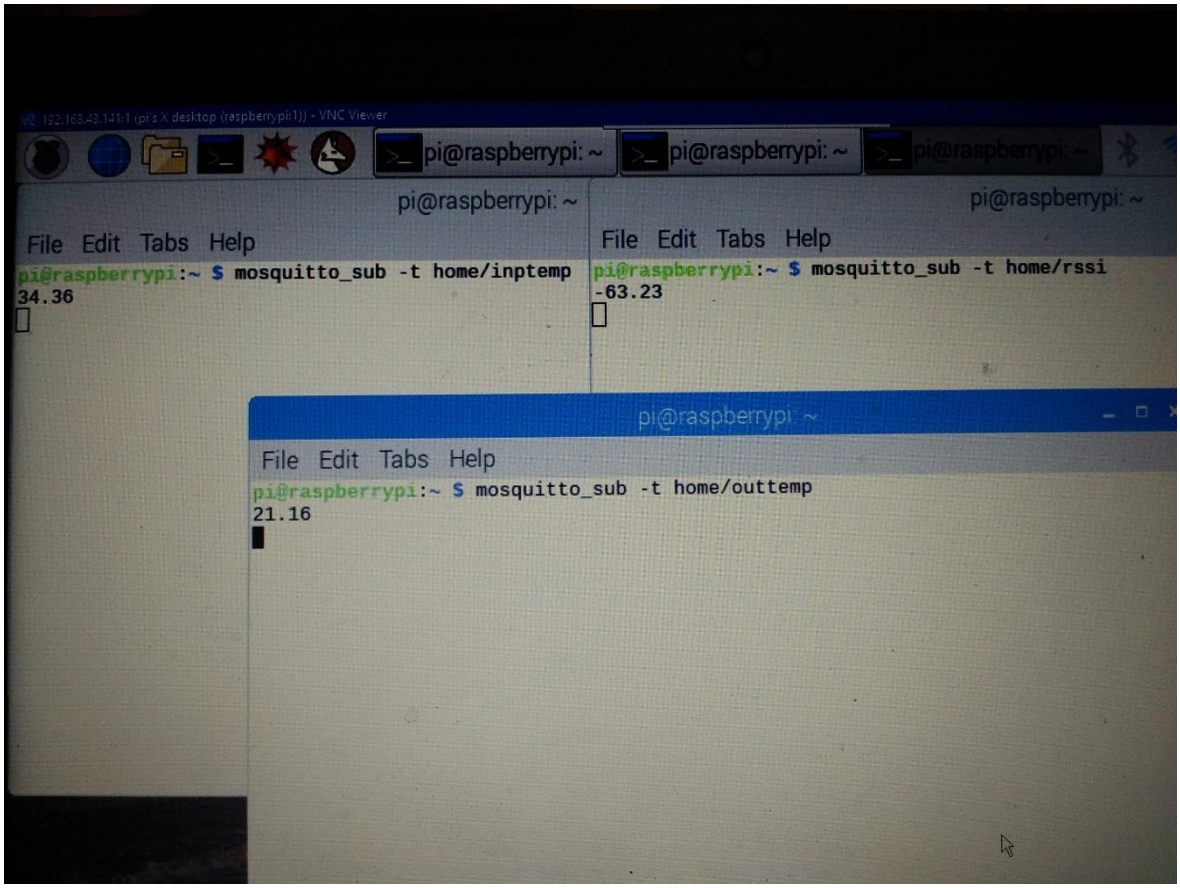

FIGURE 5.2 ABNORMAL VALUE SUBSCRIBED OVER MQTT

CHAPTER 6

SUMMARY

The main focus of this project is to help the gateway node of an IoT network to identify its faulty and malicious nodes by using a neural network model. Once faulty nodes are identified it is notified to user as an mail. Thus deploying of IoT network in complex environments is possible with in-built security.

4.1 WORK DONE IN PHASE I

- Literature survey on various IOT security threats and methods to resolve it are identified and compared.
- Validation of sensor values in gateway node, using the proposed neural network model is implemented.
- Using resilient back propogation algorithm error factor of the neural network is minimized.
- The collected data is notified through mail.

4.2 WORK DONE IN PHASE II

The contribution of this work is to formulate a neural network model specific to critical application in a complex environment with greater accuracy.

- To develop a hardware setup to sense temperature and RSSI value and communicate it to broker through MQTT.
- Train neural network on real time data fetched from edge devices to detect faulty and malicious node.
- To train neural network on multiple sensors and validating individual devices and publish invali data through mqtt to broker.

4.3 FUTURE SCOPE

The proposed model is capable of identifying malicious nodes and attacks in a network. But in controlling mechanism after detection is not automated.In future it can be improved by controlling devices on its own and including that to this model can be efficiently deployed in industrial based automation systems.

REFERENCES

[1] Aloul, and I. Zualkernan., R. Mahmoud, T. Yousuf, F., Internet of things (iot) security: Current status, challenges and prospective measures. In 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST), pages 336–341, Dec 2015.

[2] Amir shahzad A , Young-Gab Kim , Abulasad.Elgamoudi , (2017) , “Secure IoT Platform for Industrial Control Systems” , 2017 International Conference on Platform Technology and Service (PlatCon)

[3] Alanazi S., J. Al-Muhtadi, A. Derhab, and K. Saleem, "On resilience of Wireless Mesh routing protocol against DoS attacks in IoT-based ambient assisted living applications," in International Conference on e-Health Networking, Application & Services, 2015.

[4] Linlu L., "Comparative Study on the Development of IOT(Internet of Things)Policy in China and European Union Based on and," Sci-Tech Information Development & Economy, 2014.

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

[6] Navita Agarwal , Prachi Agarwal , (2013) , “Use of Artificial Neural Network in the Field of Security” , MIT International Journal of Computer Science and Information Technology Vol. 3, No. 1, Jan. 2013, pp. 42–44.

[7] Nian Xue, Lulu Liang, Xin Huang, Jie Zhang. POSTER: A Framework for IoT Reprogramming. 12th EAI International Conference on Security and Privacy in Communication Networks (SecureComm 2016), Guangzhou, China, 10-12 October, 2016.

[8] Michy Alice. Fitting a neural network in r; neuralnet package, 2015. https://www.r-bloggers.com/ fitting-a-neural-network-in-r-neuralnet-package/.

[9] Gartner Research. Gartner says 6.4 billion connected things will be in use in 2016, up 30 percent from 2015, 2015.http://www.gartner.com/newsroom/id/3165317.

[10] Juniper Research. ‘Internet of things’ connected devices to almost triple to over 38 billion units by 2020, 2015. http://www.juniperresearch.com/press/press-releases/iot-connected-devices-to-triple-to-38-bn-by-2020.

[11] Freddy K Santoso, and Nicholas C H Vun (2015), “Securing IoT for Smart Home System”, IEEE International Symposium on Consumer Electronics (ISCE).

[12] Huang X., P. Craig, H. Lin and Z. Yan. “SecIoT: a security framework for the Internet of Things”. Security and Communication Networks, 2015.

[13] Xu R., X. Huang, J. Zhang, Y. Lu and G. Wu. “Software Defined Intelligent Building”. International Journal of Information Security and Privacy (IJISP), 9(3): 84-99, 2015.

[14] Zhao K. and L. Ge, "A Survey on the Internet of Things Security," in International Conference on Computational Intelligence & Security, 2013,pp. 663-667.

[15] IoT Analytics. Why the internet of things is called internet of things: Definition, history, disambiguation. https://iot-analytics.com/internet-of-things-definition/.

[16] http://gekkoquant.com/2012/05/26/neural-networks-with-r-simple-example/

[17] https://www.r-bloggers.com/in-depth-introduction-to-machine-learning-in-15-hours-of-expert-videos/