Bachelor's thesis

Information and Communications Technology

2020

Raj Kumar Tiwari

SENTIMENT ANALYSIS OF ENGLISH TEXT USING RASPBERRY PI

– Natural Language Text Processing, Raspberry Pi, Sentiment Analysis

TURKU AMK
TURKU UNIVERSITY OF APPLIED SCIENCES

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Information and Communications Technology

2020 | 29

Raj Kumar Tiwari

SENTIMENT ANALYSIS OF ENGLISH TEXT USING RASPBERRY PI

– Natural Language Text Processing, Raspberry Pi, Sentiment Analysis

IoT devices have been used in most of the areas in daily life these days. IoT devices are the same devices used for single or multipurpose works in the home, offices, factories, industries, machinery. The data collected from the operations of such devices are usually collected and analyzed to generate further productivity and aid. Previously, the data were analyzed by humans and calculators. Nowadays, the evolving technology and programs help to understand data patterns and generate new algorithms which in turn provide more sophisticated assistance with a more advanced approach.

An example of product development and its process is explained in this thesis to understand the basic structure of IoT devices using natural language text processing to create a basic understanding of the topic. For this purpose, an embedded device was developed including a branch of machine learning to recognize the sentiment of English Text.

The thesis aims to give a conceptual overview and explain the process itself. The analysis of free text information comments and feedback helps to improve services and understand people's attitude towards the service with the help of a small processing unit such as Raspberry Pi. The device use case is aimed at customer-oriented services and it aims to collect the necessary feedback and analyze it. The device created is a basic structure that can be further modified and developed using models such as face recognition. The thesis explains the whole process of development and presents the result.

KEYWORDS:

IoT Device, Machine Learning, embedded system, NLTP, text processing, Basic concept of modern computing

CONTENTS

LIST OF ABBREVIATIONS

1 INTRODUCTION **6**
1.1 Research Background 6
1.2 Research Methodology 7
1.3 Research objectives 7
1.4 Thesis statement 8
1.5 Thesis structure 8

2 CONCEPT OVERVIEW **9**
2.1 Embedded System 9
2.2 IoT (Internet of things) 10
2.3 Network and Web Hosting 10
2.4 Natural Language Text Processing and Sentiment Analysis 11
2.5 Tools 12

3 PRODUCT DEVELOPMENT **14**
3.1 Problem and Requirement 14
3.2 Environment set up and Installation 15
3.3 Development Methods and process 18
3.4 Database 20
3.5 Apache 24
3.6 Testing 25

4 DISCUSSION **26**

5 CONCLUSION **27**

REFERENCES **28**

FIGURES

Figure 1. Raspberry PI set up with the interface 16
Figure 2. GUI of the Raspbian operating system 17
Figure 3. Display Label 19
Figure 4. Command-line interface to take input..... 19
Figure 5. Input Validation Error Message 20
Figure 6. Analyzer code that evaluates the input 20
Figure 7. Python string values to the database 20
Figure 8. Database storing the data from the python script 21
Figure 9. MySQL Login 21
Figure 10. Database created for the first-time 22
Figure 11. Select the database and create a table and columns 23
Figure 12. Verify table creation 23
Figure 13. Python data configuration 24
Figure 14. PHP connection and data fetch method 24
Figure 15. File structure from the GitHub 25
Figure 16. Apache Web display 25

TABLES

Table 1. Project Analysis 26

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
API	Application Program Interface
Apt	Advance Package Tool
ASIC	Application-Specific Integrated Circuits
CPU	Central Processing Unit
CSS	Cascading Style Sheets
DSP	Digital Signal Processors
FPGA	Field-Programmable Gate Arrays
HTML	Hyper-Text Markup Language
HTTP	Hyper-text Transfer Protocol
I/O	Input/ Output
IoT	Internet of Things
MySQL	My Structured Query Language
NLTP	Natural Language Text Processing
PHP	Hypertext Preprocessor
RAM	Random Access Memory
RDBMS	Relational Database Management System
SARP	Sentiment Analyzer Raspberry Pi
SD	Secure Digital

1 INTRODUCTION

1.1 Research Background

Technology today has gone very far, advanced, with ease of access, and solved more complex challenges with more efficiency, better and faster. With more exploration and research, the growing sophisticated needs of humanity have driven the technology into new realms such as AI, Embedded devices, web, and software programs.

Naturally, the development has become overwhelming at times for so many people to adapt to new technologies and technical terms, features, works, new programming languages, frameworks, and hardware devices. If we see it in terms of understanding and using, more specifically adapting along with trends and technologies, such adaptations have become complicated and confusing to consumers, users, academics, geeks, students, and people from different areas.

The thesis aims to provide a necessary underlying concept of the IoT device and Natural Language Text Processing (NLTP) by building a full-fledged example product. The developed example product is a web server built upon Raspberry Pi and Raspbian Linux distribution, which provides sentiment analysis of the text, based on NLTP as an application. The developed product has been termed as SARP (Sentiment Analyzer on Raspberry Pi) hereafter. The SARP is a byproduct of the process of investigating the scope of an embedded device to run machine learning applications as web services.

With the advancement in technology, embedded devices are also having different inputs of signals and data that have growing use by integrating such data with algorithms to perform complex and distant tasks. The collected text data as inputs from the user or machine can be helpful with natural language text analysis to create easier updates and facilitate the organization or owner. A large amount of feedback and/or comments on a service point and social media cannot be analyzed and calculated manually to take out the needed output. In this place, the Raspberry Pi device with the NLTP model comes into action to understand the sentiment pattern, calculate, and count the constant output from the generated input pattern.

The machines work smarter, faster, and more exact and help to automate the machines. Text processing works to fulfill the bridge gap between the human processing of the text to machine processing.

1.2 Research Methodology

This thesis is based on developmental research methods supported by knowledge acquired from my previous academic learnings and technical experience of web development, machine learning programming, and embedded systems. The theoretical part consists of a description of tools and technology from different sources, as mentioned in the references. The specific device with a similar application was not found; that is why the thesis is explorative from the perspective of development. The data used in the development are the random English sentences any subject speaks or the comments and feedback from the internet pages for the testing purpose.

The Raspberry Pi was used because of its low cost, small size, and computing capability. The software tools are chosen as per the compatibility of the need, task, and the functionality of the product. The thesis work is primarily developmental research, which aims to investigate the scope of IoT devices and NLTP for providing text processing services as a web application. Developmental research is preferred research methodology when researchers build a thing to address the given research problem (Regmi, 2019). Therefore, developmental research methodology is used for this thesis work. This thesis tries to address the research problem of investigating the scope of IoT devices to implement a machine learning application by developing a prototype device.

1.3 Research objectives

This thesis's primary purpose is to provide a basic overview of the NLTP implemented in an IoT device. Another objective is to help understand the concept, tools, and platforms needed and used for the product development for the general readers and novice students in the related field. The IoT device developed in this thesis lay a foundation of a full-fledged IoT device that can be further developed with more features like image processing, sound processing, text-processing for various purposes, and IoT related possible topics. Precisely, the thesis also explains the terms - Embedded System, IoT, Network and Web hosting, and Natural Language Text Processing.

1.4 Thesis statement

Sentiment analysis is the interpretation and classification of emotions (positive, negative, and neutral) within text data using text analysis techniques. The primary purpose of this thesis is to use the Natural language text processing in Raspberry Pi and fetch the data to the web, which can be further accessed via the Internet. In general, the reader of the thesis will have a basic understanding of the IoT and an application of Machine learning.

1.5 Thesis structure

Chapter 1 introduces the thesis work and explains the key terms in an organized manner used in the example product. Also, it establishes the connection between the intention of the thesis writer and actual work. Chapter 2 describes concepts like Embedded System, IoT, Network, and Case study.

Chapter 3 details the process of product development, such as the requirements analysis, setting up and installing the environment to start the development work, methods, and process of development and testing. Chapter 4 has documented the discussion, whereas Chapter 5 concludes the overall thesis writing procedure, product development, and results.

2 CONCEPT OVERVIEW

2.1 Embedded System

General-purpose computers such as laptops and desktops are engineered to manage a wide range of processing tasks, thus are not cost-efficient options to use where limited processing power is required to perform specific tasks only. Embedded systems are dedicated computers designed for a specific function or functions. Like general-purpose computers, embedded systems are also a combination of hardware and software that may be fixed in capability or programmable for a range of tasks. (What is an Embedded System? - Definition from Techopedia, 2016)

The size, cost, power, consumption, reliability, and performance can be optimized by engineers while designing.

Embedded system hardware

Embedded system hardware can be microprocessor- or microcontroller-based. The integrated circuit plays a very significant role in real-time computing operations. (What is an Embedded System? - Definition from Techopedia, 2016). The microcontroller that is used here is Raspberry Pi. Microcontrollers consist of CPU, memory, flash memory, RAM, and serial communication ports, whereas microprocessors constitute a central processing unit. Microcontrollers can implement low powered full systems and are frequently used on difficult and dedicated tasks.

Embedded System Software

Embedded systems are usually used as microcontrollers to carry out more straightforward computations within less memory-intensive program environments. Embedded systems often use operating systems with basic functionalities such as Rasbian for Raspberry Pi or tailored programming language platforms such as Arduino IDE for Arduino and custom minimal operating system. Generally, programs and operating systems are stored either on flash memory or rewritable flash memory. (What is an Embedded System? - Definition from Techopedia, 2016). The Raspberry Pi has used the Raspbian operating system, which is a Linux distribution. Raspbian is a Debian Linux distribution ported to Raspberry Pi. On top of that, the task design to carry out the necessary input and analysis is written on python, PHP, and web framework languages.

2.2 IoT (Internet of things)

The network of embedded devices and other microcontrollers, when connected to the Internet, is known as the Internet of Things (IoT). Here the things not only include IoT devices such as embedded systems but also IoT enabled physical assets such as vehicles with GPS trackers. (The Internet of Things (IoT) - essential IoT business guide, 2020).

IoT refers to the devices connected to the web, but the definition has been changing over time. Nowadays, IoT refers to the smart devices that communicate with each other, collecting and sharing data across the Internet. IoT devices provide an additional level of intelligence to the device handling and processes without human intervention every time. For Example, A refrigerator can check the stock of the food and order it to the grocery store. (What is embedded system? - Definition from WhatIs.com, 2020) (Ranger, 2020)

2.3 Network and Web Hosting

Computer networks and the World Wide Web are the central infrastructures of information and communications technology (ICT). The term 'web hosting' is used to indicate the process of deploying web content on the Internet. It includes maintaining a web server to store web content and managing the configuration of the necessary technology to make the stored content available via the Internet. Some of the necessary configurations are registering a domain name and associating it with a unique IP address so that the web content can be accessible in a web address. (What Is Website Hosting?, 2020)

Web server stores data in any form and serves data in the form of (HTML) web pages for client browsers. In this project, we have deployed a website using the localhost network provided by the local router. Apache webserver hosts the files within the same network in this project.

2.4 Natural Language Text Processing and Sentiment Analysis

NLP is a branch of artificial intelligence that studies interactions between computers and human languages. Among various applications of artificial intelligence, NLP deals with applications that require computers to process human languages. Such applications include speech recognition, content summarization, content translation, spam detection. (Yordanov, 2018)

The smartphone with speech recognition uses NLP to understand what is being said. Also, many people use laptops whose operating system has built-in speech recognition.

Sentiment analysis, a new field in Natural Language Processing, aims at identifying the intent of the content, and classifying opinions and sentiments. “Sentiment analysis studies people's sentiments, opinions, attitudes, evaluations, appraisals, and emotions towards services, products, individuals, organizations, issues, topics, events, and attributes. The text is classified on two different bases, such as polarity of the sentiment and polarity of the outcome."(D'Andrea, Ferri, Grifoni and Guzzo, 2015).

Different classification approaches and tools are used for sentiment analysis implementation.The Sentiment analysis is a process involving several steps such as Data Collection, Text preparation, Sentiment detection, Sentiment classification and displaying the result. Data collection from different user generating points such as content from blogs, forums, social networks, and customer-oriented service points. This raw data collected from primary sources are not organized, unstructured or in a complex language like slangs, streets, communities varying in contextual meaning. The manual analysis of the data is not a very good idea or is considered a traditional way. So, the raw data is to be cleaned and prepared so that the machine understands it easily. Therefore, text analytics and natural language text processing are used to extract and classify. Non-textual, irrelevant and stop words are eliminated from the cleaning process. The rest of the data are examined on a subjective or objective basis. Subjective sentences are classified in positive, negative, good, bad; likes, dislikes, but classification can be made by using multiple points. Sentiment analysis turns unstructured text into meaningful information. After the analysis, the test results are displayed on tables, percentages, wording (Happy, Neutral and Sad), graphs like a pie chart, bar chart and line graphs. (D'Andrea, Ferri, Grifoni and Guzzo, 2015).

2.5 Tools

Python

Python language is a high-level programming language with extensive use in a different context of modern computing. Python is an object-oriented, interpreted language with dynamic semantics. It is high-level built-in structures, dynamic typing, and binding makes the application development process easier and faster. Python reduces the cost of program maintenance thanks to its simplicity and easy to learn syntax. Python also supports modules and packages. It is open source and freely distributed. (What is Python? Executive Summary, 2020).

Due to its features and efficiency, the python programming language has been used extensively in machine learning. Several developers have contributed to its huge support of libraries and functions. The analyzer that has been used in this project uses python language.

Vader Sentiment Analyzer

"VADER (Valence Aware Dictionary and Sentiment Reasoner) is a lexicon and rule-based sentiment analysis tool specifically attuned to sentiments expressed in social media." (vaderSentiment, 2020). VADER uses a combination of lexical features that generally analyze the words with their semantic orientation as Happy and sad.

It has produced impressive outputs analyzing texts from reviews, comments, social media, product reviews, and newspaper editorials. It is available in python language that is used in this project. Vader can identify the sentiment, describes how polarized the sentiment is, and provides the score. (Staff, 2020).

PyMySQL Connector

PyMySQL is a library used to interact with the database library. It is a driver required to connect the python language script to the database.

MYSQL Server

MySQL is an open-source relational database management system (RDBMS) that helps to separate and store data in tables for software applications. MySQL database uses Structure query language to communicate with the database and has a client-server architecture. Databases are stored in files that provide optimal speed. MySQL database server is fast, easy, scalable, reliable, and works simultaneously in computers alongside other applications. (Regmi, 2019).

MariaDB is installed for creating, storing, and serving the data from and to the database.

Apache

Apache is an open-source and widely used web server software. Apache is fast, reliable, secure, free, and runs on 67% of all web servers in the world. Extensions and modules are used to highly customize the program to meet the needs of services running of different environments. (Pandey, 2018).

PHP

PHP (Pre-HyperText Processor) is installed to handle the server-side. PHP is a server-side scripting language that allows generating web pages dynamically and programmatically by allowing one to embed code within HTML templates (Regmi, 2019). The PHP script connects to the database and displays the input and Results in a webpage.

HTML and CSS

HTML and CSS are used to design the Graphical User Interface. HTML creates the page template design and frame. CSS provides the customized and required styles to the template page. PHP processes the data and generates HTML pages.

Rufus

Rufus is a tool that helps to write the memory stick with iso image files of operating systems. Rufus is used for writing the file system image containing the operating system and other applications to the SD card in this project.

3 PRODUCT DEVELOPMENT

3.1 Problem and Requirement

Research problem: The research problem is to investigate the scope of an embedded device to run machine learning applications as web services.

Research goal: The goal is to develop a prototype product implementing NLTP on Raspberry Pi.

Use case scenario:

Machine learning applications on embedded devices provide cheaper alternatives for enterprises and business to use such services in a cost-efficient way. The importance of machine learning implementation in factories, industries, and corporations to identify correct person, flawless product classification pushing the computation to the edge has been advancing and widespread. The sentiment analysis of text helps to understand the social phenomenon and bring more clarity of human communication and understanding. This kind of device can be helpful in monitoring classes with teachers, students, hospitals, and shopping mall feedback machines and can be modified with more sophisticated machine learning algorithms. The device software requires English text as input to analyze its sentiments.

Requirements,

Hardware: stand-alone IoT device. Chosen Raspberry Pi.

Software: OS, application to run a web server, application to perform sentiment analysis of the text.

3.2 Environment set up and Installation

Setup

The necessary hardware such as Raspberry Pi, screen, keyboard, mouse, and HDMI Cable was collected, to begin with, the setup process. The 16GB memory SD card from the Raspberry Pi is written with a popular Raspbian operating system using Rufus. Rufus is a popular memory writing software in a wide variety of memory sticks (MySQL :: MySQL 8.0 Reference Manual :: 1.3.1 What is MySQL?, 2020).

After the writing process is done, the SD card is inserted into the Raspberry Pi slot. The Raspberry Pi is connected to the screen via HDMI cable. The keyboard and mouse are connected to the Raspberry Pi USB slots. The power is connected to monitor and Raspberry Pi. Then the Raspbian operating system is shown on the monitor. The first part of the set up ends here.

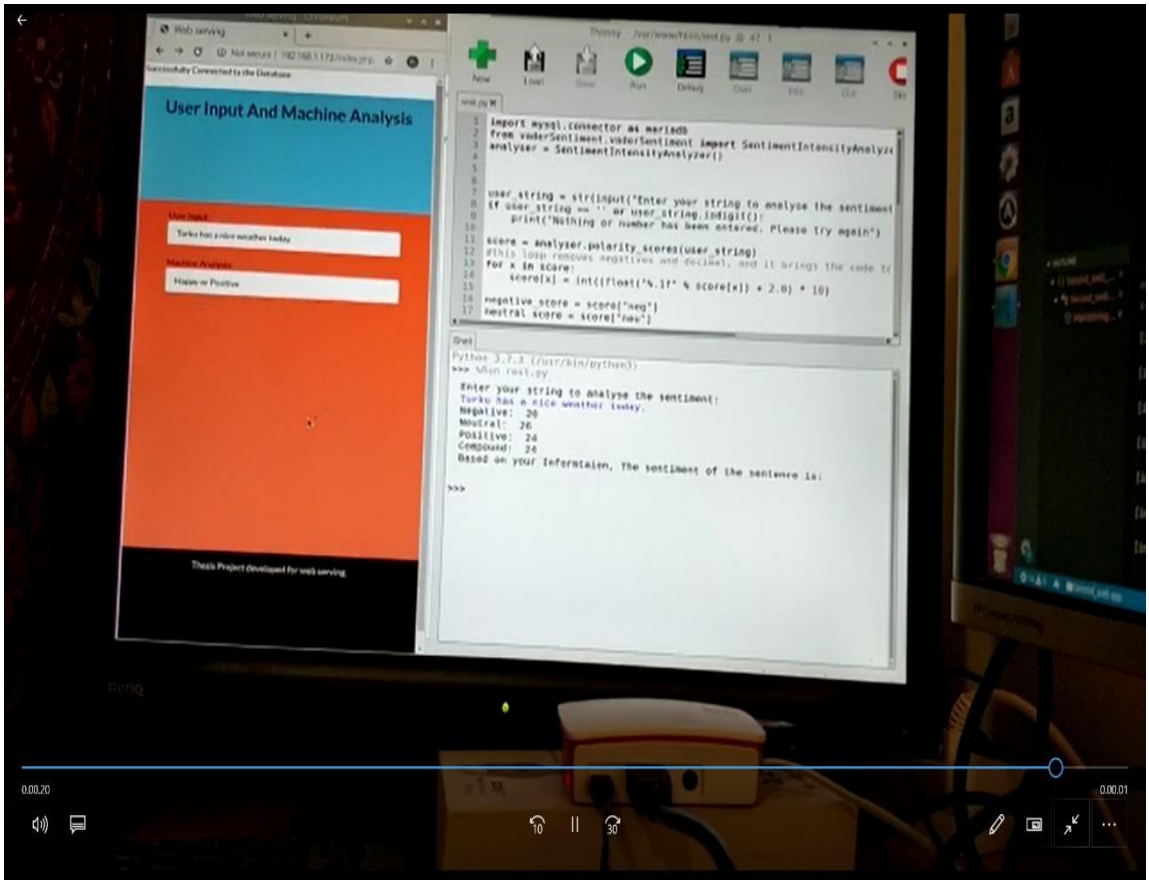

Figure 1. Raspberry PI set up with the interface

Installation

The Raspbian operating system provides the GUI and command line support to install software like Apache, MySQL, Python, and other required libraries. We prepare the development environment after installing the necessary tools mentioned above.

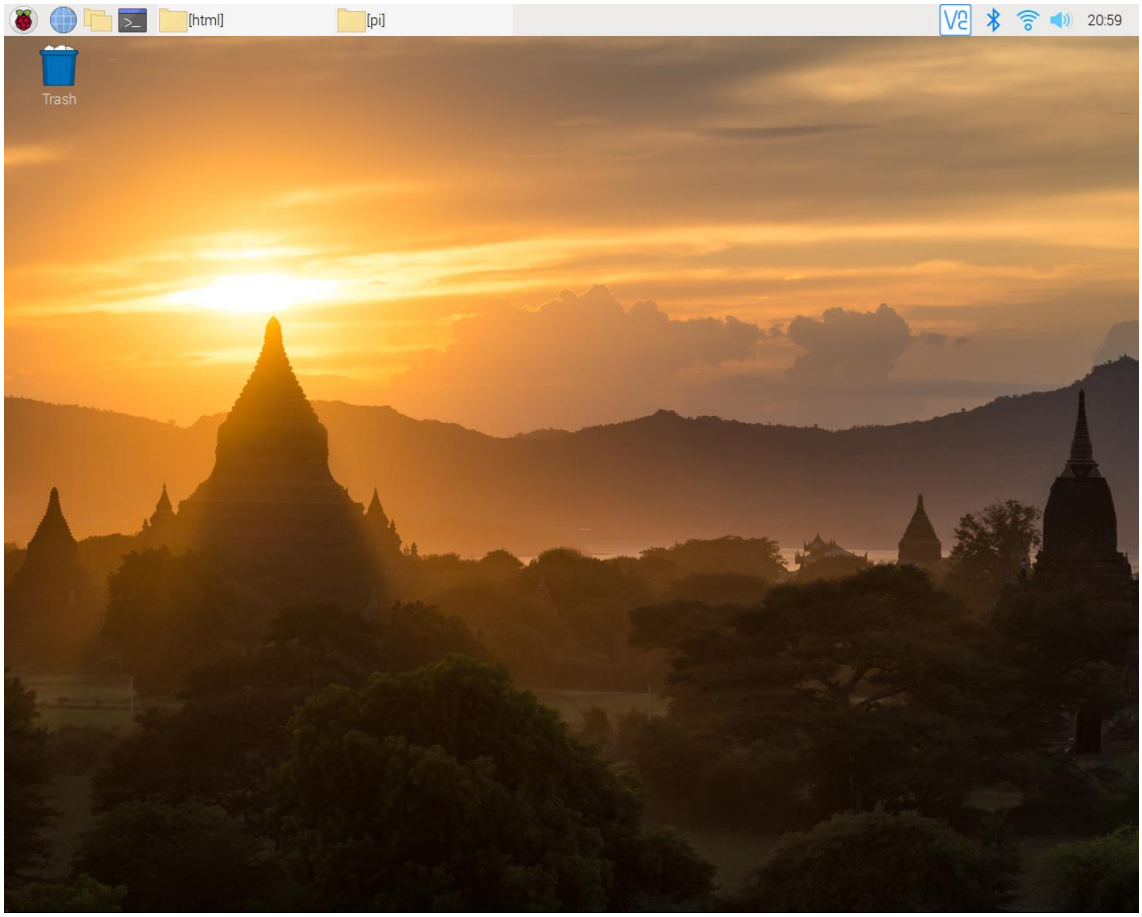

Figure 2. GUI of the Raspbian operating system

Steps:

The terminal can be found on the upper-left corner forth in figure 2.

Inside the Terminal.

Command to insert:

Apache:

sudo apt install apache2 -y

Python:

pip install MySQL-python

Vader Sentiment Analyzer:

sudo pip3 install vaderSentiment

PyMySql Connector:

sudo apt-get -y install python-mysql.connector

MariaDB Server:

sudo apt-get install MariaDB-server

PHP

sudo apt-get install PHP

After the installation, the development environment is ready to code and develop software.

Graphical User Interface

The GUI is designed to display the result that is stored in the database. HTML and CSS are used to design the front end. The GUI has two display labels, each of which displays the user input and the analyzed result.

Figure 3. Display Label

3.3 Development Methods and process

The development of SARP consist of python and PHP script, and one style files. All the program files are stored inside the var/www/HTML folder. The script file “rest.py” does the task of taking input from the user.

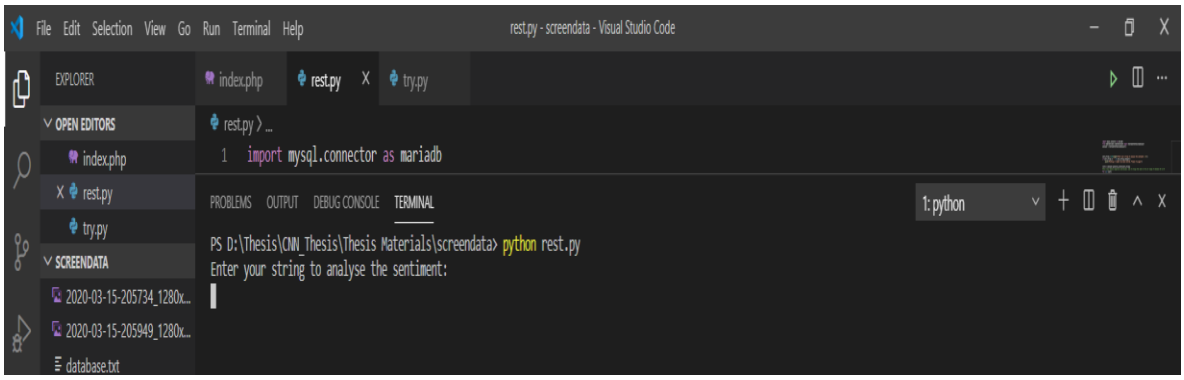

Figure 4. Command-line interface to take input

The input is validated to check if it is empty or number. It gives the error message ("Nothing or number has been entered. Please try again").

```
import mysql.connector as mariadb
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
analyser = SentimentIntensityAnalyzer()



user_string = str(input("Enter your string to analyse the sentiment: \n"))
if user_string == '' or user_string.isdigit():
    print("Nothing or number has been entered. Please try again")

score = analyser.polarity_scores(user_string)
#this loop removes negatives and decimal, and it brings the code to the int range of between 10 to 30
for x in score:
    score[x] = int((float("%.1f" % score[x]) + 2.0) * 10)

negative_score = score["neg"]
neutral_score = score["neu"]
positive_score = score["pos"]
compound_score = score["compound"]
print("Negative: ", negative_score)
print("Neutral: ", neutral_score)
print("Positive: ", positive_score)
print("Compound: ", compound_score)
print("Based on your Informtaion, The sentiment of the sentence is: \n")
```

```
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\Thesis\CNN_Thesis\Thesis Materials\screendata> python rest.py
Enter your string to analyse the sentiment:
8756287350839
Nothing or number has been entered. Please try again
```

Figure 5. Input Validation Error Message

If the input passes the validation, the input is assigned to variable user_string. The user_string is then passed as a parameter to analyze the polarity of the value. The Vader sentiment analyzer checks the polarity and intensity of the user_string and stores the result in the score variable.

```
user_string = str(input("Enter your string to analyse the sentiment: \n"))
if user_string == '' or user_string.isdigit():
    print("Nothing or number has been entered. Please try again")

score = analyser.polarity_scores(user_string)
#this loop removes negatives and decimal, and it brings the code to the int range of between 10 to 30
for x in score:
    score[x] = int((float("%.1f" % score[x]) + 2.0) * 10)

negative_score = score["neg"]
neutral_score = score["neu"]
positive_score = score["pos"]
compound_score = score["compound"]
print("Negative: ", negative_score)
print("Neutral: ", neutral_score)
print("Positive: ", positive_score)
print("Compound: ", compound_score)
print("Based on your Informtaion, The sentiment of the sentence is: \n")
if compound_score <= 17:
    results = "Sad or Negative"
elif compound_score > 17 and compound_score <=21:
    results = "Neutral"
else:
    results = "Happy or Positive"
```

Figure 6. Analyzer code that evaluates the input

The result is then measured on the 100 scales. The result is identified as Negative, Sad, and Happy or Positive derived from the resulting scale. The result is then saved to the database.

```
sql = "UPDATE nltk_results SET UserInput = %s, Results = %s WHERE id = 1;"
val = (user_string, results)
cursor.execute(sql, val)
mariadb_connection.commit()
#creating a function
```

Figure 7. Python string values to the database

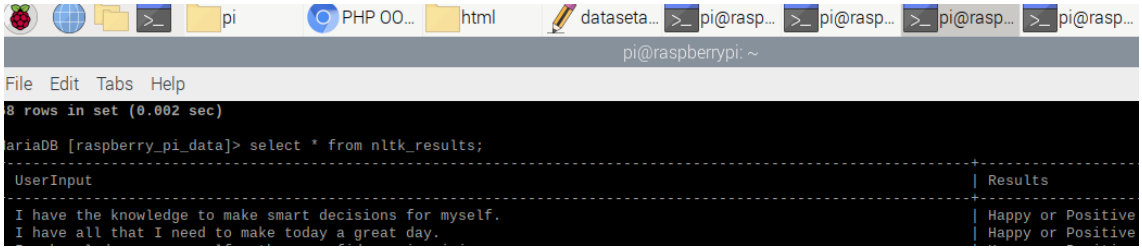

Figure 8. Database storing the data from the python script

The configuration method is discussed in the Database configuration chapter.

3.4 Database

The database called raspberry_pi_data with a table name nltk_resullts is created with two columns. The User_input and Results are the two columns inside the table.

The prerequisite installation for MySQL is completed in the installation section. The login process is mandatory to get into the MySQL Command line.

The following command is used in the terminal.

$ sudo mysql -p root -u

It required a password that we set during the installation. After the password is entered, the MySQL command line is accessed.

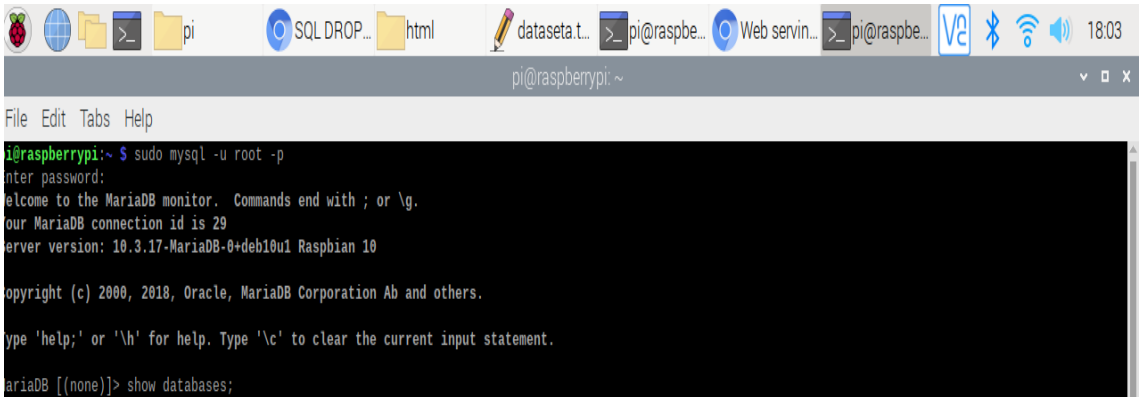

Figure 9. MySQL Login

Here, the database is created, accessed, and other operations can be performed.

The basic MySQL command is to show databases to check the existing database. This command gives the result of the existing databases.

> *show databases;*

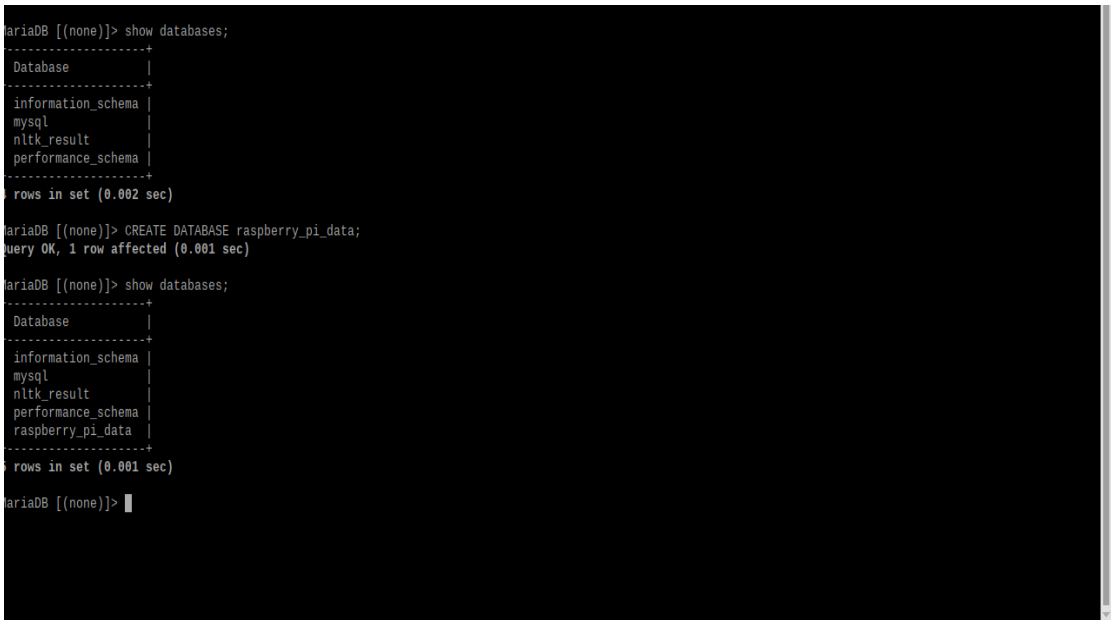

Figure 10. Database created for the first- time

In Figure 10, the result list is missing raspberry_pi_data. In the next line, the database is created with the following command.

> *CREATE DATABASE raspberry_pi_data;*

As Figure 10 displays, the database has been created.

The database needs tables and columns to store different kinds of data the program is going to need. This project requires a single table with two columns.

Firstly, it is required to select the database and execute the further operation of creating tables.

Select the database

> *use raspberry_pi_data;*

And create the tables:

```
ariaDB [(none)]> use raspberry_pi_data;
atabase changed
ariaDB [raspberry_pi_data]> CREATE TABLE nltk_results(
    -> id int,
    -> UserInput varchar(255),
    -> Results varchar(255)
    -> );
uery OK, 0 rows affected (0.070 sec)
```

Figure 11. Select the database and create a table and columns

The id is int type, UserInput and Results are the varchar type. These are the kind of data types that are used in programming to separate the nature of data.

The following command can verify the table is created or not.

*> select * from the nltk_results;*

```
ariaDB [raspberry_pi_data]> use raspberry_pi_data;
atabase changed
ariaDB [raspberry_pi_data]> select * from nltk_results;
mpty set (0.031 sec)

ariaDB [raspberry_pi_data]>
```

Figure 12. Verify table creation

The result is an empty set because there is nothing in the database. The data comes after the teaching of the language python and PHP configures with the database.

Database Configuration

The PHP script and python script use the same database raspberry_pi_data and both languages have their way to connect to the database.

Python connection method:

```
Go  Run  Terminal  Help                    try.py - screendata - Visual Studio Code
index.php   rest.py   try.py
try.py > ...
import mysql.connector as mariadb
from io import open
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
analyser = SentimentIntensityAnalyzer()

mariadb_connection = mariadb.connect(host='localhost',
                                     user='root',
                                     password="Rapassword",
                                     database="raspberry_pi_data"
                                     )
cursor = mariadb_connection.cursor()
```

Figure 13. Python data configuration

Python uses a pymysql connector to connect with the database. The mariadb.connect function has the parameter host, user, password, database, and values assign to it. The same variable is executed with the cursor function. It is the python configuration method.

After the database is connected, several SQL operations such as creating, reading, putting, updating, and deleting can be performed with the database.

Php connection method:

```
index.php ×
index.php > ...
1  <?php
2
3      //Database credentials
4      $servername = "localhost";
5      $dusernamе = "root";
6      $dpassword = "Rapassword";
7      $dbname = "raspberry_pi_data";
8
9      try {
10             $connect = new PDO("mysql:host=$servername;dbname=$dbname", $dusernamе, $dpassword);
11             // set the PDO error mode to exception
12             $connect->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
13             echo "Successfully Connected to the Database";
14         }
15     catch(PDOException $e) {
16             echo "Connection failed: " . $e->getMessage();
17           }
18     $userQueryString = "SELECT UserInput, Results FROM nltk_results";
19     $queryHandle = $connect->prepare($userQueryString);
20     $queryHandle->execute();
21     $row = $queryHandle->fetch();
22     $user_input = $row['UserInput'];
23     $results = $row['Results'];
```

Figure 14. PHP connection and data fetch method

PHP does the same task of configuration with the above code. The database credentials are assigned to the PHP variable, which then is again passed as a parameter to PDO Object inside the try-catch function. The connection method in PHP tries to connect and return Connection failed error in case of a fail database connection.

After the database is connected, the database can be accessed, and operations such as creating, reading, delete, putting, and updating can be performed from the code.

The PHP code finally fetches the data from the database with the help of the SELECT statement in Figure 4, from connection and fetch method.

Below is the file structure from Github.

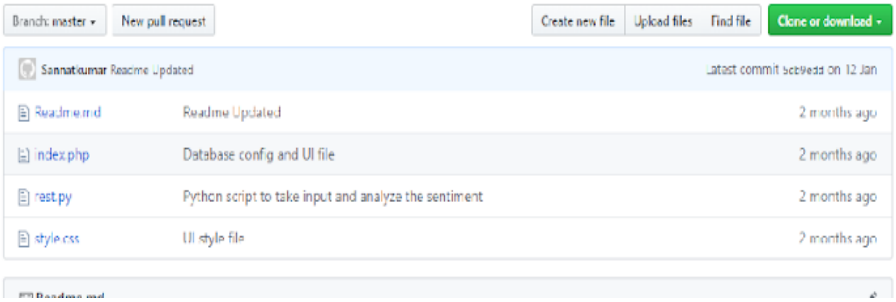

Figure 15. File structure from the GitHub

This is a single-page interface with a PHP code that configures the connection between the database and brings the data to the user in the web interface.

3.5 Apache

Finally, All the files are stored inside the var/www/html directory. The apache is started with the following command.

$ sudo service apache2 restart

After the apache is started, the page is available to be hosted on localhost. In this project, the localhost address is 192.168.1.172. The index.php is added at the end of the address to retrieve the file.

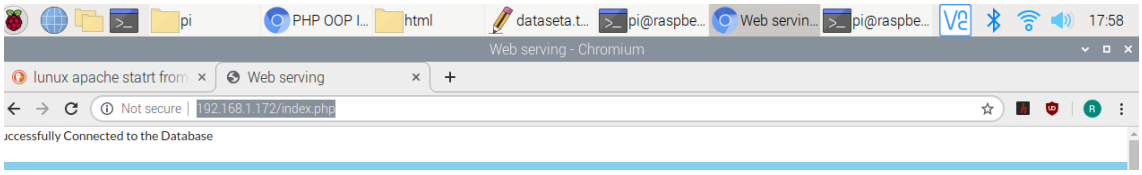

Figure 16. Apache Web display

The page is ready to serve the results.

3.6 Testing

The testing is done, and the SARP as hardware is fully functional to this task. The software has been tested with seventy-five number of statements in two datasets. The seventy-five number of datasets vary in sentiment and randomly chosen. The test results are shown below in the table.

Table 1. Project Analysis

Number of Datasets	75
Positive	48
Negative	3
Neutral	24
Accuracy	96.5%

From the Result analysis table, the analyzer has proven to be 96.5% accurate to identify the input within the test data. The critical note to always keep in mind is that the result is relative to this test data and can be different with different varieties of data.

4 DISCUSSION

Results

The result of the product development is the device name "SARP," which takes the English text as input and displays the sentiment of the input to the user and on the web. The SARP takes the user input and categorizes the input into three types of sentiment and output the compound results.

The three types of sentiment are Negative, Neutral, and Positive. These results are calculated based on the polarity and intensity of each word in the input. The result produced by the SARP has an accuracy of 96.5% with the testing dataset.

Use Case and Benefits

The SARP is a kind of feedback collecting and analyzing device where users can input text, paragraph, and files of English literature and get the result as Happy, Negative, and Normal. This device can be used in hospitals to get customer feedback where it does the task of storing and displaying the results to necessary service owners.

"The World Wide Web is an immense collection of linguistic information that has in the last decade gathered attention as a valuable resource for tasks such as machine translation, opinion mining and trend detection, that is, Web as Corpus." (Kilgarriff and Grefenstette, 2003). It can be used to analyze such data from the web, in Markets, Malls, Hotels, shops, cafeteria to gather feedbacks for the construction and monitoring of the service. Moreover, the module is the prepared underlying architecture that can be used with different other modules. The SARP helps people live and work smarter and get a better understanding between people, service customers, and create better results. The SARP provides companies' systems work, delivering insights into everything from individual performance to users.

5 CONCLUSION

In this chapter, we will discuss the conclusion based on our research and development. The purpose of the thesis was to clarify the concept and development process via theoretical research and product development. The goal was achieved through different intermediate processes of research, literature review, tutorials from different sources, and practical implementation. The final output was the product which can take any English statement as input text, process it via natural language text processing module, sends it to the internet, and can be accessed via a web user interface. During the process of writing and development, the fundamental structure was a full-fledged device. The Command-line works as the user input interface. When the script rest.py is run, then the user can input any text or paragraph. After the input is submitted, the script analyzes the given input sentiment and store the results in the database. The result is displayed on a webpage.

This thesis introduced an embedded device module that can analyze the sentiment. It introduces tools such as Rufus, Raspbian operating system, PHP, Python, MySQL, Apache web server, and the methods to use them. The SARP successfully demonstrates that the product can take input and display the result on a web page. The thesis also provides a theoretical description of the tools and technologies used for product development.

REFERENCES

A. Mishra, D. Patil, N. Karkhanis, V. Gaikar and K. Wani, "Real time emotion detection from speech using Raspberry Pi 3," 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, 2017, pp. 2300-2303, doi: 10.1109/WiSPNET.2017.8300170.

D'Andrea, A., Ferri, F., Grifoni, P. and Guzzo, T., 2015. Approaches, Tools and Applications for Sentiment Analysis Implementation. *International Journal of Computer Applications*, 125(3), pp.26-33.

Dev.mysql.com. 2020. *Mysql :: Mysql 8.0 Reference Manual :: 1.3.1 What Is Mysql?*. [online] Available at: <https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html> [Accessed 18 April 2020].

GraziCNU, M., 2017. [online] Turning Your Raspberry Pi Into a Personal Web Server. Available at: <https://www.instructables.com/> [Accessed 20 March 2020].

IoT Agenda. 2020. *What Is Embedded System? - Definition From Whatis.Com*. [online] Available at: <https://internetofthingsagenda.techtarget.com/definition/embedded-system> [Accessed 25 February 2020].

i-SCOOP. 2020. *The Internet Of Things (Iot) - Essential Iot Business Guide*. [online] Available at: <https://www.i-scoop.eu/internet-of-things-guide/> [Accessed 15 March 2020].

Kilgarriff, Adam & Grefenstette, Gregory. (2003). Introduction to the Special Issue on the Web as Corpus. Computational Linguistics. 29. 333-347. 10.1162/089120103322711569.

Networksolutions.com. 2020. *What Is Website Hosting?*. [online] Available at: <http://www.networksolutions.com/education/what-is-web-site-hosting/> [Accessed 1 March 2020].

Pandey, P., 2018. *Simplifying Sentiment Analysis Using VADER In Python (On Social Media Text)*. [online] Available at: <http://Medium.com> [Accessed 10 March 2020].

PyPI. 2020. *Vadersentiment*. [online] Available at: <https://pypi.org/project/vaderSentiment/> [Accessed 20 May 2020].

Python.org. 2020. *What Is Python? Executive Summary*. [online] Available at: <https://www.python.org/doc/essays/blurb/> [Accessed 5 March 2020].

Ranger, S., 2020. *What Is The Iot? Everything You Need To Know About The Internet Of Things Right Now | Zdnet*. [online] ZDNet. Available at: <https://www.zdnet.com/article/what-is-the-internet-of-things-everything-you-need-to-know-about-the-iot-right-now/> [Accessed 27 February 2020].

Staff, A., 2020. *What Is Apache? - What Is A Web Server?*. [online] WPBeginner. Available at: <https://www.wpbeginner.com/glossary/apache/3> [Accessed 7 March 2020].

Techopedia.com. 2016. *What Is An Embedded System? - Definition From Techopedia*. [online] Available at: <https://www.techopedia.com/definition/3636/embedded-system> [Accessed 7 February 2020].

Yordanov, V., 2018. *Introduction To Natural Language Processing For Text*. [online] Available at: <http://Medium.com> [Accessed 1 March 2020].