



REPUBLIC OF TÜRKİYE
ALTINBAŞ UNIVERSITY
Institute of Graduate Studies
Information Technology

**IOT SYSTEM FAILURE TEST USING MACHINE
LEARNING (LINEAR REGRESSION)**

Aya Ayad Hussein AL-ZUHAIRI

Master's Thesis

Supervisor

Asst. Prof. Dr. Ayça Kurnaz TÜRKBEN

Istanbul, 2022

IOT SYSTEM FAILURE TEST USING MACHINE LEARNING (LINEAR REGRESSION)

Aya Ayad Hussein AL-ZUHAIRI

Information Technology

Master's Thesis

ALTINBAŞ UNIVERSITY

2022

The thesis titled IOT SYSTEM FAILURE TEST USING MACHINE LEARNING (LINEAR REGRESSION) prepared by AYA AYAD HUSSEIN AL-ZUHAIRI and submitted on 13/12/2022 has been **accepted unanimously** for the degree of Master of Science in Information Technology.

Asst. Prof. Dr. Ayça Kurnaz TÜRK BEN

the Supervisor

Thesis Defense Committee Members:

Asst. Prof. Dr. Ayça Kurnaz
TÜRK BEN

Department of Software
Engineering,
Altınbaş University

Asst. Prof. Dr. Abdullahi Abdu
IBRAHIM

Department of Computer
Engineering,
Altınbaş University

Asst. Prof. Dr. Serdar KARGIN

Department of Biomedical
Engineering,
İstanbul Arel University

I hereby declare that this thesis meets all format and submission requirements of a Master's thesis.

Submission date of the thesis to the Institute of Graduate Studies: __/__/__

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Aya Ayad Hussein AL-ZUHAIRI

Signature

DEDICATION

First of all, I would like to thanks to Almighty God who assist me to make this thesis work. Secondly, I would like to lengthen my thanks to the esteemed authorities of Turkey for internet hosting me on its lands, then prolong my thanks and perception to ALTINBAŞ University, and to my Supervisor Asst. Prof. Dr. Ayça Kurnaz TÜRK BEN who supply me the possibility to write about such an fascinating topic. Also I would like to commit this work to my household particularly my mam and my associate Mohammed for usually assisting me and standing by means of my side. and my little princess my daughter Maryam I love you all extra than something additionally to my buddies joody and ruba , I am invariably grateful for you all .

PREFACE

The work described in this thesis has been funded by Altınbaş University Office of Scientific Research Projects as part of the Project No.



ABSTRACT

IOT SYSTEM FAILURE TEST USING MACHINE LEARNING (LINEAR REGRESSION)

Al-Zuhairi, Aya Ayad Hussein

M.Sc., Information Technology, Altınbaş University,

Supervisor: Asst. Prof. Dr. Ayça Kurnaz TÜRK BEN

Date: 12/2022

Pages: 82

In the age of the Internet of Things (IoT), day-to-day objects are outfitted with sensors and actuators. It is also referred to as IoT units that gather information and function moves through speaking with every other. Electrical units with constrained sources may additionally be subjected to big stresses or exterior influences that may additionally lead to their failure. In provider environments security is a imperative requirement, failure ought to be averted earlier than it motives any damage. By imposing predictive upkeep (PdM) the use of laptop gaining knowledge of (ML), computer getting to know algorithms can be utilized to predict screw ups and grant ample renovation time. IoT gadgets gather statistics that can be used to educate desktop gaining knowledge of algorithms to apprehend failure patterns of character devices. This thesis introduces (Decision Tree, Logistic Regression, k-nearest neighbors, Random Forest, and Gradient Boosting) algorithms and its addendum to predict the failure of electrical gadgets by means of the usage of Internet of Things devices.

The enterprise seeks to harness the blessings of integrating IoT options into each day operation to acquire clever manufacturing as nicely as clever maintenance. In fact, the value of protection is already a giant section of the budget.

Despite the integration of IoT in the industry, the value is nevertheless excessive due to the complexity of the options developed. In this paper, we suggest an IoT-based failure prediction answer designed to minimize preservation cost. Our thought lets in for relatively correct prediction of failure that leads to fantastic motion when it is clearly needed. Furthermore, we can decrease the fee of the answer itself by way of optimizing records switch between sensor nodes and the server. We got outcomes that tested the effectiveness of our answer in distinguishing symptoms of failure with no prior studying of failure and in monitoring mild deviation in laptop behavior. The proposed gadget graph allowed us to minimize packet load that would limit sensor no.de facts transmission cost, sensor node electricity consumption as nicely as community site visitors.

Keywords: Internet of Things, Device Failure, Artificial Intelligence, Machine Learning, Decision Tree.

TABLE OF CONTENTS

	<u>Pages</u>
ABSTRACT	vii
LIST OF TABLES.....	xiii
LIST OF FIGURES.....	xiv
ABBREVIATIONS.....	xvi
LIST OF SYMBOLS.....	xix
1. INTRODUCTION.....	1
1.1 MOTIVATION	2
1.2 PROBLEM STATEMENT.....	3
1.3 AIM AND OBJECTIVE.....	3
1.4 CONTRIBUTION	4
1.5 THESIS STRUCTURE.....	4
2. LITRETURE REVIWO	6
2.1 INTRODUCTION	6
2.2 LITRETURE REVIWO.....	6
2.3 CONCLOUSTION	13
3. THEORETICAL BACKGROUND.....	14
3.1 INTRODUCTION	14

3.2	INTERNET OF THINGS (IOT).....	14
3.3	IOT FRAME WORK.....	15
3.4	IOT ARCHITECTUER	16
3.4.1	Three-Layer and Five-Layer IOT Architecture.....	17
3.4.2	Cloud and Fog Based Architecture	20
3.5	IOT PROTOCOL.....	21
3.5.1	Data Link Layer	21
3.5.2	Network Layer Protocols	22
3.5.3	Transport Layer Protocols.....	22
3.6	ERROR DEDECTION	23
3.7	PREDECTIVE MAINTENANCE.....	23
3.8	ARTIFICIAL INTELLIGENCE.....	24
3.9	ARTIFICIAL NEURAL NETWORK	26
3.10	MACHINE LEARNING.	27
3.11	FAILURE PREDICTION USING MACHINE LEARNING.	28
3.12	ML ALGORITHMS	30
3.12.1	Decision Tree	30
3.12.2	Logistic Regression	31
3.12.3	K-Nearest Neighbors	32

3.12.4 Random Forest	33
3.12.5 Gradient Boosting.....	33
4. SYSTEM IMPLEMENTATION	35
4.1 INTRODUCTION	35
4.2 PROJECT TOOLS.....	35
4.2.1 Hardware Description	35
4.2.2 Software Description.....	38
4.3 DATASET	40
4.4 METHODOLOGE.....	42
4.5 PROJECT IMPLEMENTED AND RESULT	43
4.5.1 Result of Decision Tree classifier.....	44
4.5.2 Result of Logistic Regression clasifier.....	45
4.5.3 Result of k-nearest neighbors classifier.....	46
4.5.4 Result of Random Forest	47
4.5.5 Result of Gradient Boosting	48
4.6 RESULT SUMMARY.....	49
4.7 COMPARISON TO RELATIVE WORKS	50
5. CONCLUSION AND FUTURE WORK	52
5.1 CONCLUSION.....	52

5.2 FUTURE WORK.....	53
REFERENCES	54
APPENDIX A.....	64
APPENDIX B.....	66

LIST OF TABLES

	<u>Pages</u>
Table 4.1: Python Libraries used in Project	39
Table 4.2: Tested devices	41
Table 4.3: The elements of the evaluation process (variables, definitions, and equations)	44
Table 4.4: Decision Tree classification report.....	45
Table 4.5: Logistic Regression classifier report	46
Table 4.6: K-Nearest neighbors classifier report.....	47
Table 4.7: Random Forest classifier report	48
Table 4.8: Gradient Boosting classifier report.....	49
Table 4.9: Results summary	49
Table 4.10: Comparison to relative works.....	50

LIST OF FIGURES

	<u>Pages</u>
Figure 3.1: Internet of things	15
Figure 3.2: A (three layers) and B (five layers).....	18
Figure 3.3: IoT architecture shows the differ.ent layers and corres.ponding protoc.ols invo.lved in them The five layers	19
Figure 3.4: A smart IoT gateway's fog architecture	20
Figure 3.5: Comparison of RM, PM and PdM	24
Figure 3.6: A graphic depicting a neural network	27
Figure 3.7: Classification of ML algorithms	28
Figure 3.8: Failure predication model	30
Figure 4.1: Pin description of ESP32	36
Figure 4.2: ACS712 Sensor	37
Figure 4.3: DHT11 Sensor.....	38
Figure 4.4: Dataset collection circuit Then the dataset was transferred from the drive and stored as csv file	41
Figure 4.5: Decision Tree classifier confusion Matrix	44
Figure 4.6: Logistic Regression classifier confusion Matrix.....	45
Figure 4.7: K-nearest neighbors classifier confusion Matrix	46
Figure 4.8: Random Forest classifier confusion Matrix	47

Figure 4.9: Gradient Boosting confusion Matrix.....	48
---	----



ABBREVIATIONS

(RNNs)	:	Recurrent Neural Networks
AI	:	Artificial Intelligence
ANN	:	Artificial Neural Network
ARIMA	:	Auto-Regressive Integrated Moving Average
BN	:	Byesian Network
CEP	:	Statistic Control Process
CIOs	:	Chief Information Officers
CNN	:	Convolutional Neural Networks
CoAP	:	Constrained Application Protocol
CPoF	:	Complex Patterns of Failure
DBN	:	Deep Belief Network
DES	:	Double Exponential Smoothing
DNN	:	Distributed Network Name
DT	:	Decision Tree
FCFS	:	Fog Work Flows Simulates
FNN	:	Feedforward Neural Network

FPP	: Failure Prediction Platform
FT	: Faults Test
GA	: Genetic Algorithm
GBM	: Glioblastoma
GPS	: Global Positioning System
GPUs	: Graphics Processing Units
HVAC	: Heating, Ventilation, and Air Conditioning
IC	: Integrated Circuit
IMMS	: Intelligent Material Management System
IoT	: Internet of Things
kNN	: K-Nearest Neighbor
LAN	: Local Area Network
LGR	: Long-Range Guided Rocket
LPWAN	: Low Power Wide Area Network
LSTM	: Long Short Term Memory
MARK	: Monitor and Analyze Resource Keys
MBP	: Multi-Purpose Binding and Provisioning
ML	: Machine Learning
MLP	: Multilayer Perceptron

MQTT : Message Queue Telemetry Transport

NFA : National Futures Association

NNs : Neural Networks

NTC : Negative Temperature Component

PdM : Predictive Maintenance

PM : Preventive Maintenance

PNN : Probabilistic Neural Network

QoS : Quality of Service

RF : Random Forests

RFID : Radio Frequency Identifying Card

RFs : Random Forests

RM : Reactive Maintenance

SVM : Support Vector Machines

SVMs : Support Vector Machines

TPUs : Tensor Processing Units

LIST OF SYMBOLS

I : Current Ampere

V : Voltage Volt

P : Power Watt

T : Temperature $^{\circ}\text{C}$



1. INTRODUCTION

An (IoT) Symbol refers to Internet of Things [1]. It can be defined as a technology that allows for the autonomous communication of things, making existing environments smarter. They frequently have sensors that gather environmental data or employ actuators to execute environmental activities. A smart home can include an HV AC (heating, ventilation, and air conditioning) system. [2] to regulate the room temperature, humidity, and overall air quality. By continually monitoring the state of the space and taking the appropriate action, it gives the inhabitants comfort. To collect the necessary data, the system often uses temperature and humidity sensors. Then, based on the user-defined presets, the system will automatically turn on the heater, ventilator, or air conditioning system.

IoT gadgets may be used to keep an eye on the manufacturing machinery and processes in a Smart Factory [3]. The information gathered provides an understanding of how production is progressing and can be used to identify any anomalies in the processes or equipment. IoT devices may also be used in harsh locations, although doing so comes with significant restrictions and difficulties [4]. In the case of precision farming, the equipment might be buried to track soil moisture. Devices can be placed underwater at various depths to check the quality of the water. Extreme weather circumstances call for dealing with particular difficulties, such as electricity usage, environmental degradation, communication barriers, etc.

Environmental factors and computational load could affect IoT devices, which could significantly increase the likelihood of failure. Additionally, the limited resources can only promote various kinds of breakdowns. Failures brought on by memory fatigue, for instance, might result from approaching memory constraints [5] or from obstructions in wireless communication pathways [6].

The corresponding IoT structures regularly have safety-critical functions that name for options to end more than a few failure types. Based on prior failure predictions, Predictive Maintenance (PdM) [7], is a renovation method that allows the scheduling of upkeep schedules for a variety of sorts of equipment. Typically, the approach makes use of Machine Learning (ML) [8].

Techniques, which make use of the facts gathered from the units to analyze about and forecast probably faults. Neural Networks (NNs), additionally recognized as Artificial Neural Networks (ANNs), Support Vector Machines (SVMs), and Random Forests (RFs) are a few ML strategies that are broadly used to allow (PdM) [9], [10].

IoT setups regularly have a massive range of units with special characteristics. Even identically classified gear may additionally be subjected to more than a few exterior effects, main to quite a number failure patterns. These specific failure patterns may additionally be taught into character ML algorithms. Additionally, several algorithms would possibly be skilled on the equal data, combining their findings, if accuracy is a imperative requirement. Due to the conceivable for producing a excessive variety of models, a mannequin administration platform for failure prediction is required. IoT structures are regularly used to manipulate IoT settings by way of gathering the fundamental information from the related units that are already in use. The Multi-Purpose Binding and Provisioning (MBP) Platform is described in this thesis [11], [12], which is one such platform. In addition to the extensive functionality the MBP offers.

The thesis seeks to examine five different algorithms performance in failure prediction. The chosen algorithm must be in a position to assume IoT gadget disasters by way of gaining knowledge of from preceding IoT data. For the evaluation, an splendid measure and dataset have to be applied. Due to the a couple of obstacles of the real-world IoT data, it used to be determined to strengthen artificial datasets in order to take a look at the algorithms. Additionally, the thesis tries to recommend a reusable computer learning-based strategy for IoT gadget failure prediction. This is carried out by way of introducing a platform dubbed the Failure Prediction Platform, each as a notion and a working prototype (FPP). The platform is an surroundings the place many ML fashions may additionally be hosted and built-in as standardized modules.

1.1 MOTIVATION

In general, the world is moving towards the Internet of things in all areas of life. In our study, this thesis contributes to providing a way to solve the sudden stop of electrical devices and giving an alert to the owner of the device about the possibility of the device malfunctioning or

failing soon, as our study contributed to the employment of Internet of things and artificial intelligence in solving the problems of failures and predict it before it occurs.

1.2 PROBLEM STATEMENT

IoT devices including sensors, actuators, and embedded systems are frequently used in a smart home setup to keep track of the environment and status of the household appliances. This implies that any issues with such IoT devices might put the entire system at risk of failing and compromise the property's quality and safety.

Electrical devices have a life span granted by their manufacturers. For this reason, these devices are exposed to a decrease in the life span as a result of the increase in operating hours, and for this we proposed a way to predict the failure or failure that encounters the devices as a result of consumption or the effect of capacity. In our study, we provided a solution to predict failures and stop devices before they occur by measuring the ability of devices and using machine learning. We also proposed Decision Tree, Logistic Regression, k-nearest neighbors, Random Forest, and Gradient Boosting algorithms, where it proved its efficiency with a high value This proves the success of our work.

The research problem can be summarized in the following question, how to test failure of electrical devices by using Internet of Things devices and machine learning?

1.3 AIM AND OBJECTIVE

- i. We offer an effective and innovative method for detecting sensor drift flaws in time-series data streams using forecasting and classification using Decision Tree, Logistic Regression, k-nearest neighbors, Random Forest.
- ii. Create a tool that uses supervised learning methods to locate sensor drift problems in data streams.
- iii. Determine that Decision Tree, Logistic Regression, k-nearest neighbors, Random Forest, and Gradient Boosting are effective at detecting sensor drift problems in this particular scenario.

- iv. To apply the unique strategy that has been recommended in a cloud computing environment, analyze the response time tradeoffs.

1.4 CONTRIBUTION

Relying on the Internet of Things, we recommend a answer to predict failure to screen the machines. This answer makes use of a sensor node that consists of vibration and / or sound sensors and then blended with the proposed large statistics evaluation science to predict the failure of the machine, and hence a answer used to be observed to achieve a set of contributions related to facts and hardware analysis. Our contribution can be summarized in:

- i. Optimizing payload through sensor node for low cost data transmission.
- ii. The ability of the solution to expand by avoiding the extra effort to install the system.
- iii. Anticipate any type of failure risk, including inexperienced types, from day one.

1.5 THESIS STRUCTURE

This study contained five chapters, the first chapter contained a general introduction on the subject of the thesis and reviewed the research problem and the most important goals and motives for this study. The related works listed in chapter two. The theoretical study of the project, where it started with an introduction about the Internet of Things and its purpose, then talked about artificial intelligence and what it contains of machine learning, and then touched on the most important types of machine learning, which are (supervised learning, unsupervised learning, reinforcement learning, and finally deep learning), Finally, the chapter dealt with the maintenance of electrical appliances.

With regard to the third chapter, it examined the failure of Internet of Things devices, the faults that can occur in them, some designs for error handling and detection, and finally, the proposed Internet of Things-based error prediction system. And the fourth chapter contained the practical side of the study, where it started with the problem studied by the researcher, and the hardware and hardware components that were used in the project, then touched on the selection of algorithms and how to design and train the neural network, then reviewed the results of testing

the designed network, and finally compare the results Which were reached with some of the results of previous researchers for the purpose of showing the quality, accuracy and modernity of the proposed technology compared to the above.

Finally, the fifth chapter reviewed the most important conclusions reached by the researcher, as well as the suggestions and recommendations for subsequent researchers for the purpose of benefiting in this field.



2. LITRETURE REVIEW

2.1 INTRODUCTION

Many studies describe failure prediction techniques and highlight certain IoT devices or particular causes. Certain methods make use of machine learning algorithms, whereas others employ basic statistical techniques. ML (machine learning) Various machines' faults are predicted using approaches. The main purpose of the suggested solution is to monitor equipment in facilities and factories. Since electromechanical systems comprise the majority of the target devices that need to be monitored, monitoring of vibrations and sounds has been successful for decades.

This section focuses on IoT systems where the vibration sensor and/or microphone serve as the sensor node's central components The Internet of Things (IoT) is highly resilient and has substantial economic impact about society. The Internet of Things markets are anticipated to consist of 84 billion or more connected devices generating 186 zettabytes of data in 2025 [13], [14]. Many researchers have worked in the Internet of Things and the expected errors and failures in this system.

2.2 LITRETURE REVIEW

[D. Ganga *et. al.*, 2017] recommended employing an IoT2040 gateway and Real-time condition monitoring of electrical devices using vibration analytics [15].

[Windau *et. al.*, 2017] Introduce system for Monitoring Inertial Machines is recommended (IMMS). Sup port vec tor machines (SVM) and neural networks (NN) were employed in this study, and their classification performance was compared. The number of inertial sensors required to discriminate between ten different categories of anomalous equipment behavior in the real world was decreased by the authors [16].

[WANG *et. al.*, 2017] suggested using machine learning to monitor performance and foresee breakdowns in optical networks. Sup port Vec tor Mac hine (SVM) and Do uble Expo nential Smoothing are the fundamental algorithms for this approach (DES). The suggested protection method focuses on ri sk-aware mo dels in opti cal networks and mainly investigates how to

forecast equipment failure risks. To our knowledge, this crucial issue hasn't been adequately taken into account. According to the experimental findings, the method employed to predict the breakdown condition of optical equipment had an average prediction accuracy of 95. This outcome indicates that the technique can accurately forecast the likelihood of equipment failure. Consequently, the suggested DES-SVM strategy can successfully enhance conventional risk-aware models to safeguard services from probable failures [17].

[Li *et al.*, 2018] use Smartphone battery life can be predicted using tree-based machine learning (ML) models and linear regression (LR). where a set of data used to develop and evaluate the models. Other smart devices that generate data that is comparable to those produced by the proposed method can also be used. The scientists found that battery discharge history is the most crucial factor for prediction and that tree-based models outperform LR [18].

[Fafoutis *et al.*, 2018] Using the previous measurements of current consumption, determine the battery life of IoT devices. The CC26503 MCU is used to conduct the observations using wearable and environmental sensors [19].

[Rafiuzzaman *et. al.*, 2019] Describe a technique for predicting failure. He asserted that IoT device failure results from memory fatigue. In general, IoT devices run general-purpose applications that are not designed for their limitations and frequently crash due to memory fatigue. Predictions with different lead times are made the use of the k-Nearest Neighbor (kNN) method. Rafiuzzaman et al. developed the MARK (Monitor and Analyze Resource Keys) notion as a phase of the solution, which contains gathering statistics from the devices, processing it, and then giving it to the kNN-based prediction mannequin [20].

[Soualhia *et. al.*, 2019] Give the variety of IoT devices, researchers furnish a framework for fault detection and forecasting in edge cloud environments the usage of statistical and computing device getting to know strategies [21].

Additionally, they evaluated the Raspberry Pi 2 with CPU Using convolutional neural networks (CNN), forecast failure and memory stress, (LSTM), a form of Recurrent Neural Networks (RNNs), and their combinations. Based on the findings, the framework is able to quickly and

accurately identify and anticipate numerous faults. Pre-processing is another element of the architecture that, according to the authors, is essential for enhancing performance and lowering costs.

[Bhavana *et. al.*, 2019] offered an idea that included automatic fault identification and monitoring of all home appliances. This study is concerned with defect prediction in real-world gadgets. For determining whether a device is abnormal, Naive Bayes calculations are used. As a result, the suggested approach speeds up defect detection. This method is proposed, and its implementation makes use of cloud computing, IoT, and mobile apps, hence the term "multi-technology environment" for the system [22].

[Xenakis *et. al.*, 2019] suggested an IoT and cloud-based architecture for industrial IoT's monitoring of machine status and defect prediction. This study provided MATLAB-based real-time maintenance focused on the cloud. To reduce expenses [23].

[Bhatnagar *et. al.*, 2019] listed numerous ML techniques that have been found to improve speed and produce better predictions and classifications. Numerous system performance indicators, including noise reduction, data recovery, and prediction rate, are improved by this method (by utilizing device location and status to produce better and quicker outcomes.) Using clustering techniques, neural networks, and Bayesian networks, the error reduction rate is roughly 1%. Other methods that work well include support vector machines, decision trees, regression analysis, principal component analysis, k-nearest neighbors, random forests, and evolutionary algorithms [24].

[Suga *et. al.*, 2019] Explain a technique for calculating the likelihood of a QoS (Quality of Service) failure of the wifi connection. The authors contend that while communication failures are often rare during the training process, ML-based forecasts of the QoS values cannot be reliable and sufficient. The approach proposed employs the needed QoS as a threshold to assess the outage risk, together with error distribution of various rarity classes and QoS values predicted via Probabilistic Neural Networks (PNN) [25].

[Lian *et al.*, 2019] describe a solution for Internet of Things devices to monitor aging and anticipate lifetime. Integrated Circuit (IC) technology may age at a different rate in IoT devices

depending on the environment or the specific device. Lian et al. system continuously gathers data from each device, saves it in the cloud, and then processes it to provide an IC-specific ageing model. The device's lifespan can be predicted using the ageing model. Another advantage of cloud computing, according to the researchers, is the central collection of ageing data for all ICs of the same type, which makes it simple to identify anomalies [26].

[Pandey *et al.*, 2019] created a system for predicting battery life and optimizing power with an emphasis on cellphones. The battery discharge patterns of certain users serve as the primary source of data for forecasting [27].

In recent years, work has continued to predict battery life, where failure of devices can be predicted by predicting the life of their batteries, and this aimed at improving the energy in smart phones [18].

[Teoh *et al.*, 2019] introduces a genetic algorithm (GA)-based resource management system. In addition to Min Min, Max Min, FCFS, and Round Robin, Fog Work flows simulates the time, cost, and power performance of GA. Using real-time data sets, the predictive maintenance model was created using a two-class logistic regression. The outcomes demonstrate that the suggested method beats Round Robin, FCFS, Min Min, Max Min, and Min Min in terms of execution time, cost, and energy consumption. Compared to the second-best result, execution time was 0.48% faster, costs were 5.43% less, and power consumption was 28.10% less. The prediction model has a 95.1% training accuracy and a 94.5% testing accuracy [28].

[Alexander & Kotonya, 2019] suggests Complex Patterns of Failure (CPoF), a method for providing reactive and proactive FT (ML). Error detection techniques are used in all CEP systems and are referred to as non-specific finite mechanisms (NFA). To train ML models that proactively address impending future events of known faults, FT interactive support is tracked and used for learning. Through tests that predicted battery depletion failures using time and error correlations, we assessed CPoF on an internal culture system. Predictive models are taught to take advice from live FT assistance and offer proactive error recovery [29].

[Ara *et. al.*,2020] to anticipate wireless sensor node failures, use ARIMA (Auto-Regressive Integrated Moving Average). ARIMA predicts future faults by recognizing patterns in time series data. By contaminating a sensory dataset, mimicking errors, and producing particular patterns, the authors assess the model. To get the best results, the data also undergoes various pre-processing stages. The technique is shown to be more effective for short-term forecasts than long-term ones [30].

[Souri *et. al.*, 2020] Present a prediction model that uses multilayer perceptron's and particle swarm optimization for formal verification and behavioral modeling with ML. In order to validate failure prediction behavior in IoT applications, defect metrics were also found. The outcome demonstrates that, as compared to other methods, the verification method consumes incredibly minimal operational time and memory. As is obvious, the use of machine learning in IoT networks is a relatively new idea. Failures may occur as a result of wireless Internet of Things device communication [31].

[Bhavana Kotte *et. al.*, 2020] present a study say that the main focus is on foreseeing faults in practical technologies that facilitate our daily lives. Here, past faults that have been online and recovered utilizing cloud computing technologies are included in the device database. This will make it easier to anticipate issues with the upgraded hardware. It also looks for flaws in the tools using the Naive Bayes account. The model described for this article entails keeping track of every home device online to detect problems with little to no human involvement [32].

[Zhang *et. al.*, 2020] suggested a unified, block chain-based learning approach. The integrity of verifiable consumer data is enabled by the Federal Learning Systems for Fault Detection in IIoT platform design, which we first introduce. The design states that each client generates a Merkle tree on a regular basis, stores the root of the tree on the block chain, and represents each leaf node as the client's data record. Additionally, a novel center-distance-weighted average (CDW Fed Avg) technique was put out to solve the issue of data heterogeneity in the detection of IoT failures by taking into account the distance between each customer data set's positive and negative categories. Additionally, a connection-based smart incentive mechanism is developed based on the volume and central location of client data utilised to train the local model in order

to encourage clients to take part in federated learning. Together with an industry partner, we are developing a prototype of the suggested architecture and testing its performance, accuracy, and viability. The outcomes demonstrate that this strategy is workable and has acceptable accuracy and performance [33].

[Bhavana et. al., 2020] looked at predicting malfunctions in actual gadgets that improve our daily lives. The prior mistakes were retrieved online using cloud computing technology and are now included in the device database. This will make it easier to foresee issues with the optimised hardware. Additionally, it makes use of the Naive Bayes account to point out tool flaws. The model proposed for this article entails internet monitoring of every home device in order to detect issues with little to no human involvement [34].

[Ouda et. al., 2021] suggests a framework to optimize the predictive/corrective upkeep graph and predict desktop faults the use of sensor data. Machine mastering (ML) fashions are developed the usage of preceding information to forecast the failure costs over the following 5 days. A quantity of techniques are compared, along with function extraction methods, selects, and ML fashions (based on each regression and classification). We exhibit how prediction fashions can assist raise machine reliability at decreased fees through feeding the output of the computing device studying fashions into an optimization mannequin that proposes an optimized renovation strategy [35].

[Whereas Wang et. al., 2021] proposed a cooperative office automation system that boosts team productivity. This system's property of constant, reliable functioning that conserves resources was present [36].

The writers spoke about the IoT-based technologies that will suggest upcoming IoT solutions [37].

[Imeri Amil , 2021] offers the Failure Prediction Platform (FPP), a platform that makes it easy to maintain and integrate machine learning models that have been built to forecast the failure of Internet of Things (IoT) devices. The performance of failure prediction, training time, and prediction of several ML algorithms are also contrasted. According to the evaluation's findings,

linear classifiers performed poorly in terms of failure prediction, while tree methods performed the best. Finally, a time-series prediction FPP prototype was developed and shown employing long-term memory (LSTM) as an online learning strategy [38].

[Wahid et. al., 2022] suggests a framework for predicting laptop breakdowns and for enhancing the timing of preventive and corrective maintenance. The failure chances for the following 5 days are estimated the use of computing device gaining knowledge of (ML) fashions that have been skilled on preceding data. Several approaches, inclusive of characteristic extraction methods, selects, and ML fashions (both regression and classification based) are contrasted. We illustrate how prediction fashions can increase gadget reliability whilst decreasing expenses with the aid of feeding the output of the desktop studying fashions into an optimization mannequin that proposes an optimized upkeep approach.

The proposed hybrid CNN-LSTM framework is a two-stage end-to-end model in which 1-D CNNs are in fee of efficiently extracting high-level factors from the statistics and the LSTM is used to find out about the correlations amongst a wide variety of time-series information variables by its memory function. By placing aside the non-everlasting dependency patterns of severa time-series variables, our strategy learns the long-term patterns of the time series. CNN-LSTM provided the most honest and right forecast in our contrast [39].

[Mihigo et. al., 2022] introduce a small long-term reminiscence (Tiny LSTM) and a dense serial neural network, two compact real-time predictive analytics models, are brought (DNN). Utilizing the reput of the equipment's many components, the Edge Impulse Tiny Model is used to estimate the equipment's final usable life. Real-time facts gathered from operational gear was once used to consider gear degradation insights. A supervisor's expertise-based fuzzy good judgment was once utilized to categorize the dataset and outline renovation priorities, which are then used to decide the actual usable existence that is nevertheless left. With ranking losses of 0.01 and 0.11, respectively, for the LSTM and the mannequin from Edge Impulse, predictive analytic fashions are surprisingly developed and operate well. For distribution on devices, each fashions have been changed into Tiny Models. To replicate the deployment of each Tiny Models, hidden statistics used to be employed. The comparison and deployment outcomes

display the effectiveness of Edge Impulse's Tiny LSTM and Tiny Model in real-time predictive maintenance, however it is a long way less difficult to build, migrate to, and installation the Edge Impulse model to the Tiny model [40].

2.3 CONCLOUSTION

Machine learning technologies will make symmetric Internet of Things (IoT) applications among the most important new data sources in the future for predicting failures. In this context, cloud computing is given the ability to access different types of empirical symmetric data across a large number of network devices, study data information, gain knowledge, and make informed decisions based on the set of data at its disposal. Where a method has been developed to estimate the prediction of faults coming from the sensor to deal with them before they occur ,with the help of IoT based framework for predict sensor failures. Thus, we conclude from these studies several things, including:

- i. Future device failure is foreseen and prevented.
- ii. Based on the rate of replacement, the producer can forecast the quality of their products.
- iii. It is anticipated that product production would increase.
- iv. Forecast the product lifespan for a certain production plant.

3. THEORETICAL BACKGROUND

3.1 INTRODUCTION

This chapter dealt with the theoretical side of the study, which included the definition of the Internet of hardware devices, and then addressing the failure of electrical devices and its causes, as well as addressing artificial intelligence and Machine learning and its algorithms, and finally explaining the algorithm used in this study.

3.2 INTERNET OF THINGS (IOT)

We go through IoT layers, general IoT problems, and specifically the difficulties in integrating all medium-sized data center facilities side systems with the IoT platform in this section. Current IoT networks present a number of difficulties for CIOs and other analysts, according to the literature. We will explicitly cover four topics to go over the issue, the state of the art at the time, its limits, and the solution you offered.

Currently, device manufacturers sell solutions with proprietary control units such as fan coil controls, ceiling-to-unit controls, and heat pump unit controls. They address the most famous units, and leave the others. Most large equipment manufacturers have closed systems that only support their branded products [1]. The current limitation is the lack of vendor neutral gateways and consoles that handle the many configurations to connect to different peripherals in heterogeneous data centers. In order to handle connectivity to multiple peripherals in data center ecosystems.

Data center networks: Human interactions with applications in data centers require a moderate amount of bandwidth [1]. The Internet of Things promises to significantly disrupt these patterns by sending massive amounts of SMS sensor data to the data center for processing, which in turn greatly increases the bandwidth demand of the internal data center [2]. Reliable controller topologies and clustered controllers are not used in heterogeneous data centers.

The Internet of Things' (IOT) necessities can be relaxed through utilising cloud-based architecture. This approach centralizes software Genius and storage inside server records centers. However,

there should be a breakdown if there are extra requirements, such as real-time desires for the manage loop, a lot of data, or a deployment structure with inadequate community bandwidth. As a result, the idea of "edge computing" has emerged and has supported the demand for decentralized processing. The purpose of such a technique is to foster dispersed talent as nicely as an mixture of conveniently reachable sensed bodily records that resembles a gateway. Deterministic and real-time processing are required to execute specific functionalities in such a platform.

Although there is no consensus on a specific and described structure for the Internet of Things, a prevalent structure with three layers the Perception Layer, the Network Layer, and the Application Layer is extensively diagnosed.

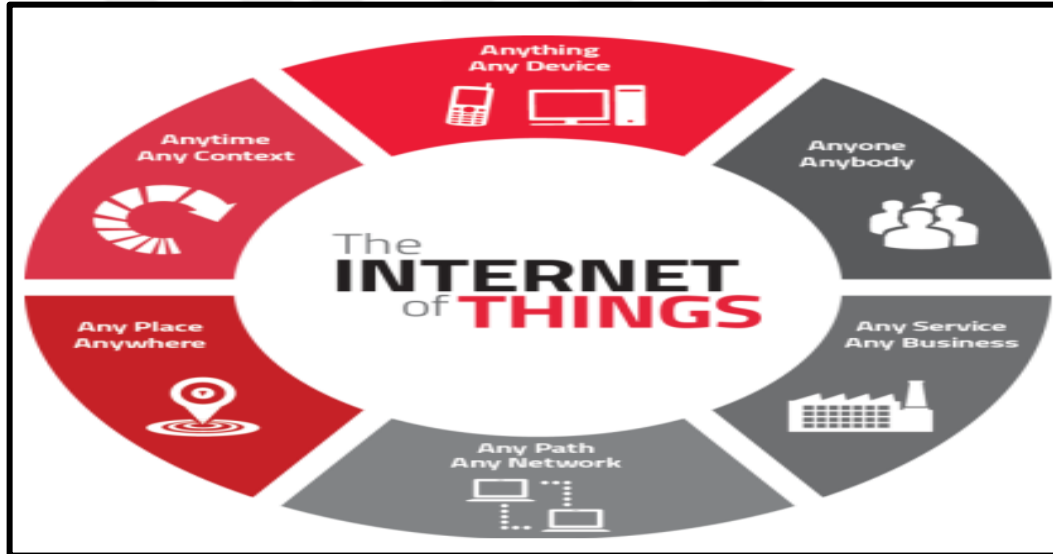


Figure 3.1: Internet of Things [43].

3.3 IOT FRAME WORK

We go through IoT layers, general IoT problems, and specifically the difficulties in integrating all medium-sized data center facilities side systems with the IoT platform in this section. Current IoT networks present a number of difficulties for CIOs and other analysts, according to the literature. We will explicitly cover four topics to go over the issue, the state of the art at the time, its limits, and the solution you offered.

Currently, machine producers promote options with proprietary manage devices such as fan coil controls, ceiling-to-unit controls, and heat pump unit controls. They tackle the most well-known units, and depart the others. Most massive tools producers have closed structures that solely assist their branded merchandise [1]. The current limitation is the lack of vendor neutral gateways and consoles that handle the many configurations to connect to different peripherals in heterogeneous data centers. In order to handle connectivity to multiple peripherals in data center ecosystems.

Data center networks: Human interactions with applications in data centers require a moderate amount of bandwidth [1]. The Internet of Things promises to significantly disrupt these patterns by sending massive amounts of SMS sensor data to the data center for processing, which in turn greatly increases the bandwidth demand of the internal data center [2]. Reliable controller topologies and clustered controllers are not used in heterogeneous data centers.

The Internet of Things' (IOT) necessities can be cozy by means of utilizing cloud-based architecture. This approach centralizes utility brain and storage inside server facts centers. However, there ought to be a breakdown if there are extra requirements, such as real-time wishes for the manage loop, a lot of data, or a deployment structure with inadequate community bandwidth. As a result, the thought of "edge computing" has emerged and has supported the demand for decentralized processing. The aim of such a approach is to foster dispersed Genius as properly as an combination of conveniently handy sensed bodily records that resembles a gateway. Deterministic and real-time processing are required to execute precise functionalities in such a platform.

Although there is no consensus on a precise and defined architecture for the Internet of Things, a well-known architecture with three layer the Perception Layer, the Network Layer, and the Application Layer is widely recognized.

3.4 IOT ARCHITECTUER

There is no single consensus on the structure of the Internet of Things, as many distinct buildings have been proposed through many researchers [44]. In general, the IoT structure consists of the devices, community topology, and cloud science that permits IoT units to communicate with

every other .The improvement of Internet of Things protocols has made it viable to join billions of bodily units round the world, as mo.re and greater gadgets are linked to Internet traffic. They turn out to be larger, require greater facts processing skills and storage life. Different units function with distinct protocols and requirements as there is no unified IoT fashionable to work with and this poses a big assignment to the contemporary infrastructure that is now not designed to take care of these extra billions of IoT devices. IoT purposes are connectivity, scalability, integration, intelligence, and detection. The effect of the Internet of Things and the integration of cloud computing need to be studied to predict failure earlier than it occurs.

3.4.1 Three-Layer and Five-Layer IOT Architecture

Although there is no agreed upon and extensively standard shape for the Internet of Things. The Internet of Things is divided The IOT gadget in to three or 5 layers, the three-layer structure is the fundamental IoT structure that consists of:

- i. Perce.ption (sensors, gad gets, and other devic es)
- ii. Network (communication betw een devi ces)
- iii. Appli.cation (the la.yer with which the user interacts)

Thr ee-layer architecture is not enough when dealing with detailed analysis of the Internet of Things. This led to Add more layers and suggest five layers and seven architectural layers. Middleware layer and Business Layer are adding on the basic three-layer architecture to become five layer.

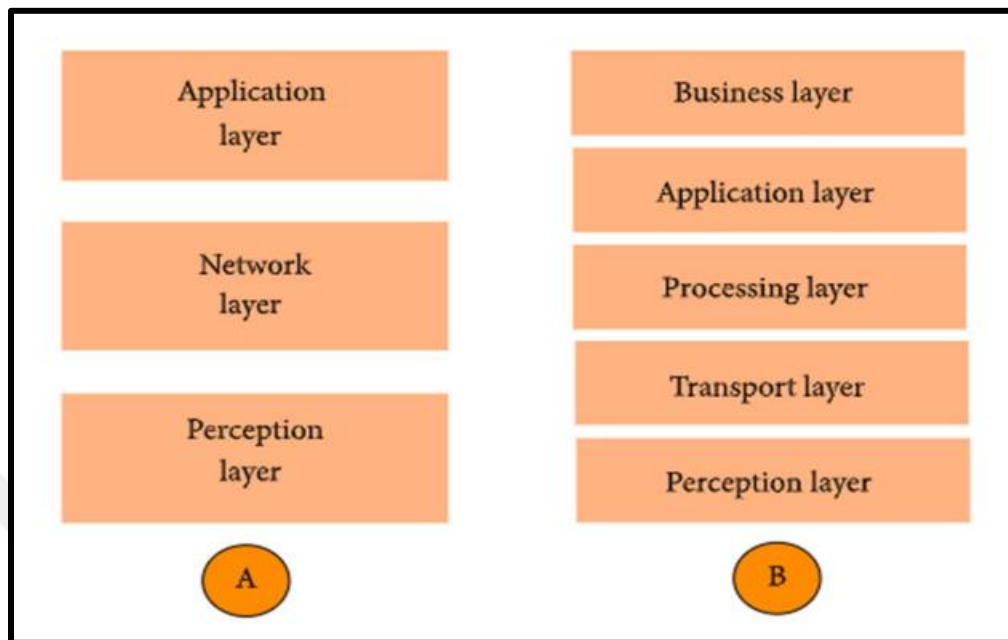


Figure 3.2: A (Three Layers) and B (Five Layers) [45].

Figure 3.2 A shows the basic three layers; perception layer, Network layer and Application layer on top of each other while Fig3.2B shows the five-layer model.

The following sections discuss the five layers of IOT architecture.

- i. **Perception Layer** The function of this layer is to collect and capture data. This layer consists of sensing units and physical devices. This layer can contain all types of sensors, sensor gateways, RFID tags, cameras, and GPS data. On this layer, objects are also uniquely identified. After this layer collects information, it is passed to the Network layer.
- ii. **Transport layer** This layer consists of the converged community infrastructure with more than a few sorts of conversation technology. This layer transmits records gathered from the Perception layer to the facts processing unit. The facts are transferred to the Middleware layer [45], Where extraordinary sorts of conversation applied sciences are used in the network layer such as Wi-Fi, Bluetooth, LPWAN, 3G , and so forth.

- iii. **Middleware Layer** The function of this layer is the management of various devices and information processing in the Internet of Things. This layer can determine which devices can or cannot communicate with each other.
- iv. **Application Layer** The important characteristic of this layer is the alternate of records securely, this layer consists of utility options that in flip construct facts technological know-how options to the entire IOT structure.
- v. **Business Layer** This layer manages the system for the entire IoT architecture. In order to make correct decision for the business model [45], the results must be carefully analyzed in this layer.

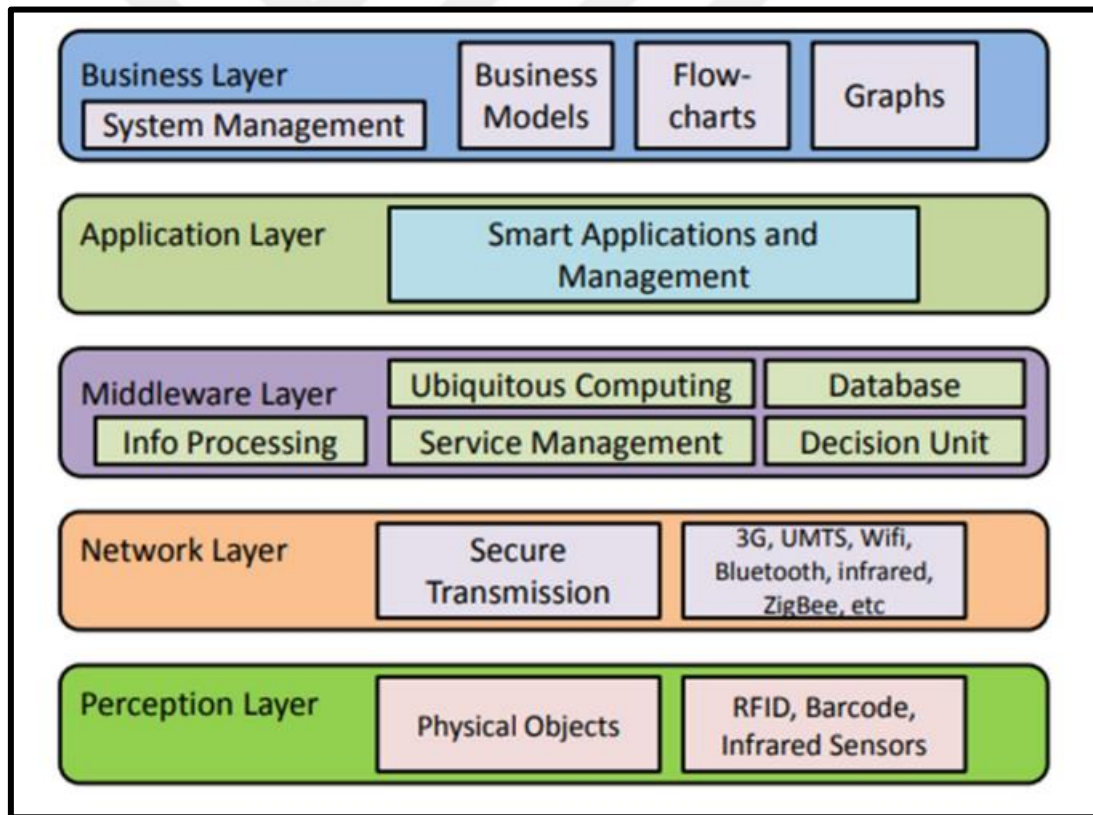


Figure 3.3: IoT architecture shows the different layers and corresponding protocols involved in them. The five layers [46].

3.4.2 Cloud and Fog Based Architecture

This structure classification to cloud and fog depends on the location of the data process unit in IOT System.

i. Cloud Based Architecture

It is a device in which records is processed in cloud computer systems that generally have a increased storage ability and processing strength [47]. the Cloud based totally structure affords the scalable options as storage and processing electricity the ability can without difficulty be adjusted primarily based on the kind and complexity of the problem. The scalability lets in this structure to follow a answer to a greater community of matters and with giant quantities of facts to be method.

ii. Fog Based Architecture

There is a go toward this structure recently. Instead of relying on far flung cloud sources [48], [49], this structure includes information processing in the neighborhood of bodily devices. Network gateway gadgets and sensors do section of the facts processing. This structure provides monitoring, preprocessing, storage, and safety layers on pinnacle of it. The bodily layer and beneath the transport layer, as proven in discern 3.4.

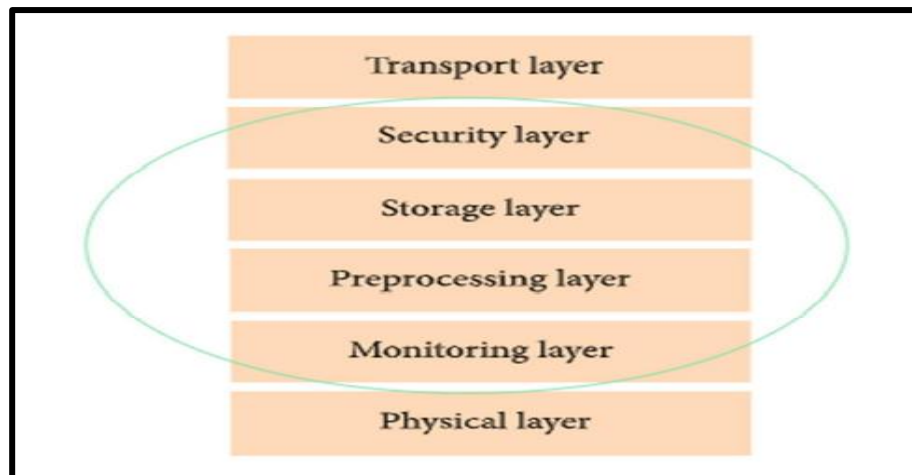


Figure 3.4: A Smart IoT gateway's fog Architecture [50].

The seven layers used in the IoT structure are proven in Figure 3.4. The monitoring layer continues of strength consumption, system aid utilization, machine responses, and a variety of services. The preprocessor layer in part filters, processes, and analyzes facts generated by means of the bodily layer. The storage layer is in cost of facts replication and storage. and The safety layer ensu.res facts confidentiality and statistics integrity.

3.5 IOT PROTOCOL

The TCP/IP model is divided into four layers, each with its own set of protocols. with different protocols used in each layer. The TCP/IP stack consists of four layers: the data link layer, the network layer, the transport layer, and the application layer.

3.5.1 Data Link Layer

This layer connects two or greater bodily devices, and a gateway system join the complete crew of units to the internet. The bodily gadgets or sensors are analogous to gadgets in a Local Area Network (LAN). The a range of units inside the LAN can speak with one some other except the want for an exterior net connection. However, if the conversation entails a system outside the LAN, the gateway gadget routes the records to the internet. Traditional records hyperlink layer protocols used with the aid of computers, smartphones, and drugs are incompatible with IOT applications. They have a massive overhead body dimension and devour a lot of power. IOT units normally have much less processing power, much less statistics storage capacity, and are predicted to run on low strength (typically batteries) for prolonged intervals of time. This requirement triggered the advent of a new set of protocols for IOT applications. IoT protocols are commonly required to have a small overhead body dimension and devour little strength [51].

Some IoT protocols at the data link layer include:

- i. IEEE 802.15.4.
- ii. IEEE 802.11 AH.
- iii. WirelessHART.

- iv. Z-Wave.
- v. Bluetooth Low Energy.
- vi. Zigbee Smart Energy.
- vii. DASH7.
- viii. Long-Term Evolution Advanced (LTE-A).
- ix. LoRaWAN.

3.5.2 Network Layer Protocols

Due to the broad adoption of IPv6 addressing, IOT devices can now have individually recognizable IPv6 addresses as IPv4 addresses become increasingly limited. However, most IOT devices won't be able to use IPv6 since its addresses are too lengthy (128 bits). New IPv6 protocols for IOT devices have been adopted as a result [51].

The following standards were developed by the IETF for IoT applications.

- i. 6 LoWPAN (IPv6 over Low Energy Wireless Personal Area Network).
- ii. 6TiSCH.
- iii. 6Lo (IPv6 over resource-limited nodes networks).

3.5.3 Transport Layer Protocols

Various corporations have proposed and applied a variety of kinds of requirements and transport protocols. TCP or UDP protocols can be used for transport in IoT applications. However, there are exceptional transport protocols designed for IoT functions that take into account the special necessities of IoT devices. The protocols have to be light enough to run on IoT gadgets with restrained memory, processing power, energy consumption, bandwidth, and value [52]. the two most extensively used IoT transport protocols nowadays are MQTT and CoAP.

3.6 ERROR DETECTION

An error is defined as a deviation of a program's operation from its exact requirements caused by the presence of bugs that appear only when the program is running or being tested [53]. An error is detected when it is indicated by an error message/signal, and it is latent when it goes undetected [54].

3.7 PREDECTIVE MAINTENANCE

Maintenance ensures that a precise gadget stays in right working order. Depending on the method used, one-of-a-kind charges are related with maintenance. There are quite a number of classifications of protection strategies, which commonly end result in three extraordinary techniques [55]:

- i. **Reactive Maintenance (RM):** Maintenance is solely carried out after the gadget has failed. It is additionally referred to as corrective protection or running-to-failure. There are no maintenance prices at the time of computing device operation, however restore or alternative charges are probable to be excessive. This commonly results in extensive machine downtime after a failure happens.
- ii. **Preventive Maintenance (PM):** Also known as deliberate or time-based maintenance, preservation is carried out at normal intervals even if the computer is in best working condition, and the time interval is decided on the groundwork of one-of-a-kind criteria. This generally leads to needless and highly-priced upkeep, however might also stop failure and related prices.
- iii. **Predictive Maintenance (PdM):** is performed at pre-determined intervals, depend on failure expectations. Forecasts are based on information previously collected through device monitoring.

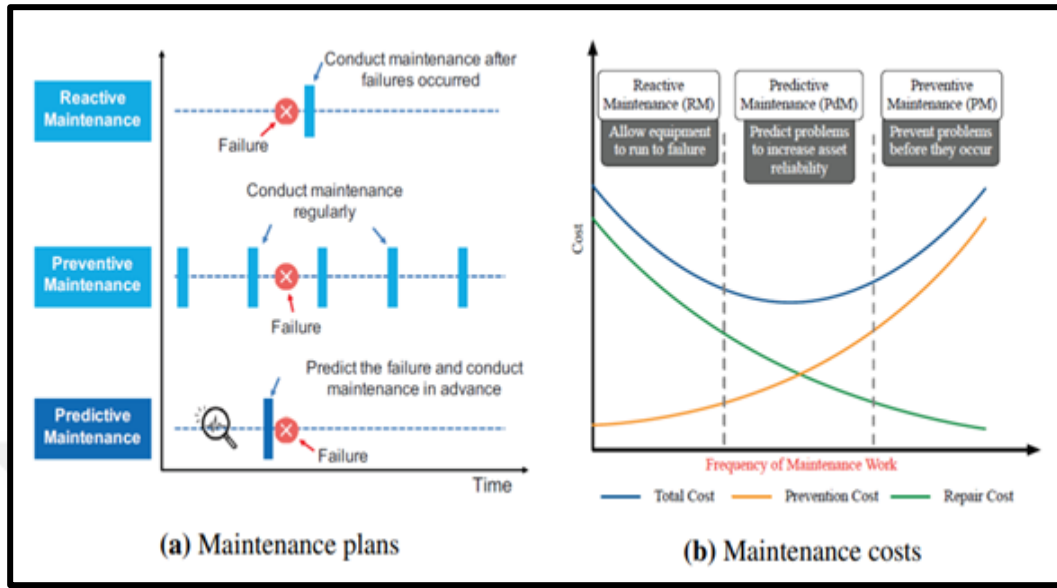


Figure 3.5: Comparison of RM, PM and PdM [56].

Figure (3.5a) indicates an overview of the a number renovation techniques with the aid of displaying Maintenance time. Figure3.5b gives the expenses related with every of them upkeep strategy. RM have excessive restore fees and low preventive costs, and vice versa Around. PdM is a stability between these two strategies, ensuing in decrease ordinary costs.

As a frequent approach, a range of ML algorithms are used to predict screw ups PdM [56]. To be in a position to function preservation and stop failure, Forecasts have to grant ample time. The time between awaiting failure and the incidence of the failure is indicated as a time-out[57]. Lead time, that is, the greater the expectation in the future, the longer renovation time Available. There are a variety of strategies of predicting failure that can shop ample time for maintenance. Classification algorithms can be used to predict machine failure inside a positive time window [58]. Moreover, time collection forecasting can be Used to predict sequences that grant records about future screw ups [59].

3.8 ARTIFICIAL INTELLIGENCE

Artificial Genius (AI) is the find out about of how to make machines sensible so they can operate jobs that have historically required the human mind. In phrases of application, adaptation, processing speed, and capacities, AI-based structures are growing quickly.

Less everyday responsibilities are an increasing number of being dealt with with the aid of machines. AI simply entails "choosing" a applicable selection at the right time, whereas human talent simply entails "taking" a best choice at the proper time. Simply said, AI lacks the decision-making inventiveness that people possess.

Although it is feasible to argue that human inventiveness will continually trade the nature of productive work, AI-based structures have pretty elegantly minimized the repetition of human efforts and can produce results in a comparatively quick quantity of time. The majority of modern-day AI lookup can be categorised as "Narrow AI." This implies that science solely improves a pick out few tasks. However, that is no longer what we are going for at all. As a result, a range of fields have mixed to increase AI.

The interdisciplinary issue of AI has been boosted via the collaboration of several fields, along with philosophy, computer science, mathematics, statistics, biology, physics, sociology, psychology, and many more. All the facts produced in every of these fields is what makes Genius possible. It is essential to analyse these facts in order to expose the underlying concepts. Although it takes a lot of time, the human Genius is effortlessly successful of reaching it. This is due to some undesirable traits of the records in the actual world:

- i. Huge volume.
- ii. Unstructured nature.
- iii. Varied data sources.
- iv. Needs real-time processing.
- v. Changes continuously.

Therefore, statistics science techniques are regularly utilized in AI. Broadly speaking, facts science is the science of developing equipment and tactics to have a look at huge quantities of information and derive insights from it. Consequently, the self-discipline is a synthesis of several different fields of study. Ideas for growing equipment are more often than not drawn from pc

science, which is chiefly involved with storage scalability and algorithmic efficiency. Ideas for evaluation come from a lots wider vary of sources.

Methodologies are taken from both the social sciences and the simple sciences, which include physics, statistics, and layout idea (like economics, sociology, political science). Data science is additionally more and more famous for sure multidisciplinary approaches, such as sample attention [60], machine learning [61], data mining [62], database management systems [63], and big data analytics [64].

Machine learning is a key method for achieving artificial intelligence (ML). A certain class of learning issues can be resolved by the human brain. For instance, the visual system contains a large number of optical neurons that facilitate human object recognition. Learning is not just limited to people; it also extends to other living things like plants and animals. A toddler learns to speak, a plant learns to adapt to its surroundings, a bird learns to soar, and so on. Our capacity to pick up new skills and adapt to our surroundings is essential to our survival. Machines can be programmed to learn, adapt, and improve themselves in a manner similar to how humans learn [65].

3.9 ARTIFICIAL NEURAL NETWORK

An artificial neural network is one of the forms of artificial intelligence that based on ML to be creative, it consists of a group of artificial neurons, each of which takes a set of inputs and then sends a specific output. ANN has a group of neurons called the input layer, and another group of neurons that make up the output layer, and then one or more hidden layers that connect between The two layers [66] as in the illustration (3.6). NN cannot be manually created. It would be ideal to utilize machine learning on the training data set, to use it for detection. After that, the model would be modified until its forecasts were sufficiently precise. In order to respect the limitations of IoT resources during this phase, be careful not to build the network too vast and complex. While neural networks can be made more effective in a variety of ways [67], [68], the truth It remains that the neural network is slow to train, and often, unless it is carefully built Requires resources [68]. At the very least, their use requires the use of a digital zoom [67]. However neural networks have been successfully employed.

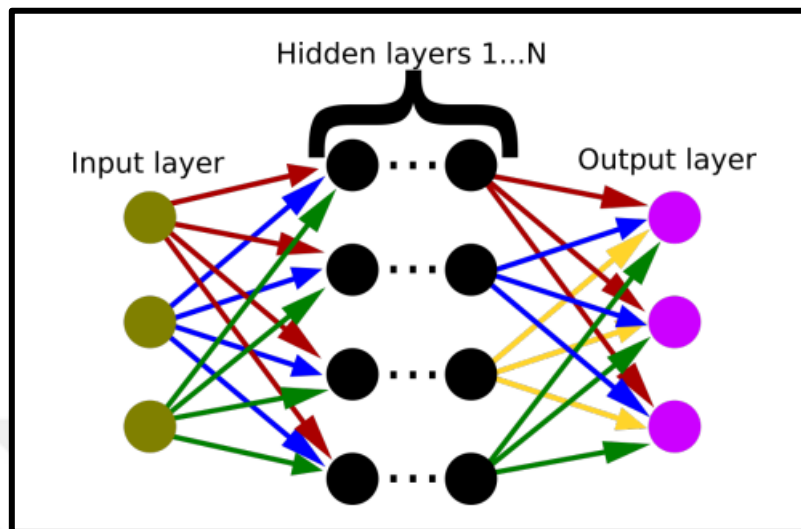


Figure 3.6: A graphic depicting a neural network.

The first of the N black hidden layers (between which input/output arrows are absent) receives its output from the dark-yellow input layer, while the final one receives its output from the purple output layer. The colorful input/output arrows serve only to make them simpler to recognize from one another.

3.10 MACHINE LEARNING

The ability of a machine, software, or algorithm to enhance performance or make predictions by learning from past data is known as machine learning [69]. The quantity and quality of the learning data influence how well ML algorithms perform. The machine learning algorithm picks up knowledge from previous data to develop an ML model for prediction. When learning online, a dynamic data set is used for both the prediction phase and the learning phase, but when learning offline, the phases are clearly separated and the data set is static [70]. There are differences between offline and online learning, as well as between lazy and motivated students. The processing of data is delayed by lazy learners until a prediction request is made. Unlike eager students who had previously analyzed data to produce a set of rules utilized, for predictions [71] Rather than simulating intelligent behavior, as is the case with artificial intelligence, machine learning is a branch of (AI) that makes use of computers.

Standard AI case [72]. There are numerous classifications of ML algorithms [73], which results in three major categories.

- i. Supervised learning: The output of the algorithm is a label so The coaching statistics is marked. The algorithm maps the enter to a predefined label. It is the most frequent Used for the classification and regression duties.
- ii. Unsupervised learning: The output of the algorithm is unknown, so the coaching facts is unlabeled the algorithm tries to discover patterns or frequent factors in the enter data. It is most regularly used for clustering duties.
- iii. Reinforcement learning: no education facts in the usual experience of education and testing, the levels appear at the identical time. The algorithm learns via interacting with the surroundings with the aid of receiving the comments in the shape of rewards.

The categories are illustrated in Figure3.7The figure also contains some algorithms used in practice for each category. In addition to the points presented there is a subcategory based on NNs [74], called Deep Learning (DL). Unlike a traditional neural network, where a deep neural network (DNNs) contains multiple layers that enable them to solve complex problems.

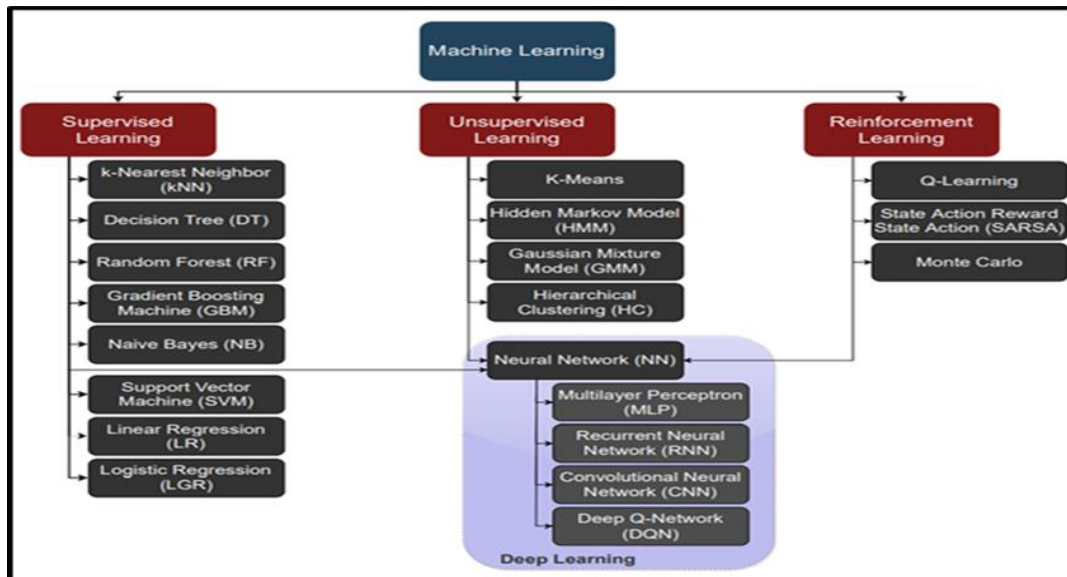


Figure 3.7: Classification of ML Algorithms.

Algorithms for machine learning can be used to address a variety of issues. ML is frequently used to forecast the condition of gadgets, machinery, and equipment or to reduce failure costs and operations to plan maintenance.

3.11 FAILURE PREDICTION USING MACHINE LEARNING

The main objective of the failure prediction model is to predict early job / task failure in cloud IoT applications more accurately by using ML classification algorithms. The proposed model helps to increase the efficiency and performance of IoT cloud infrastructure as well as reduce computational time and resource consumption. This section discusses the main components of the proposed framework for failure prediction. There are many solutions that improve the reliability and availability of cloud services, such as predicting job failures, changing priority policies, developing scheduling algorithms. For the design of the failure prediction model, there are four important stages:

- i. The cloud administration machine video display units the required assets such as CPU, memory, and disk space, after which the monitored statistics will be saved in the cloud in order to be used for the failure prediction levels.
- ii. Apply data preprocessing and the filtration techniques is necessary in data science. The quality of failure prediction model is based on this step.
- iii. To predict failed jobs or tasks, a machine learning algorithm will be applied to cloud workload tracers.
- iv. Finally, based totally on the forecast results, the cloud administration gadget will make the terrific decision. If the job is anticipated to be a success, it will be supplied and commonly scheduled to the on hand nodes. Otherwise, if the job is anticipated to fail, failure mitigation strategies will be used.

Recently, RF has been successfully applied in a variety of sequence modeling tasks [75]. In this work, we use an RF-based prediction network to describe the dynamic nature of computing systems due to the substantial temporal correlations in the aforementioned computing system

failure prediction. The failure likelihood in the near future is determined by the prediction model, with early warning signs indicating a potential failure in the future. Typically, the early warning indication is weak, making it challenging to model using basic models.. The stages of the failure prediction model are depicted in Figure (3.8).

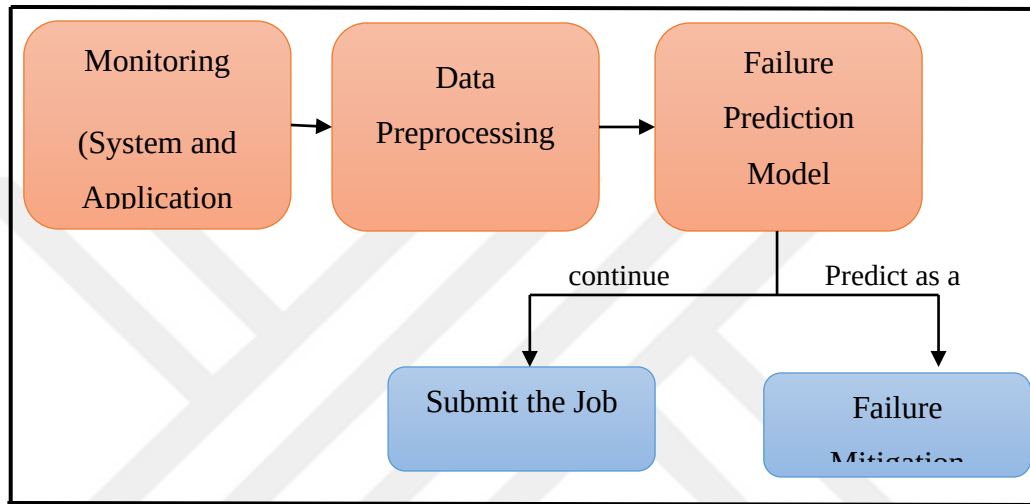


Figure 3.8: Failure predication model [76].

3.12 ML ALGORITHMS

Multiple sorts of supervised laptop gaining knowledge of classifiers exist. Naive Bayes, linear and quadratic discriminant analyses, generalized linear models, stochastic gradient descent, assist vector machines, linear help vector classifiers, choice trees, neural community models, closest neighbors, and ensemble strategies are some of these types. By combining susceptible learners, the ensemble tactics produce sturdy learners. These predictive models' primary purpose is to make bigger accuracy throughout the board.

3.12.1 Decision Tree

One of the most well-known machine learning algorithms is the decision tree classifier developed by Quinlan [77]. Decision nodes and leaf nodes are the elements of a selection tree. Each selection node is a check X over a single attribute of the enter data, and it has a variety of

branches, every of which offers with the consequences of the check X. Each leaf node symbolizes a classification that is the effect of a case's choice.

A selection tree is truly created by means of a divide-and-conquer approach [77]. T education facts units are divided into okay lessons (C1, C2,..., Ck). T will be a leaf if T completely consists of instances from a single class. T is a leaf and its related category will be assigned to the foremost category of its mum or dad node (this is the desire of C4.5). If T does now not include any cases. If T consists of examples of combined classes, or greater than one class, T will be divided into n subsets (T1, T2,..., Tn), and a take a look at based totally on an attribute of the coaching facts will be conducted. the place n is the whole quantity of take a look at consequences for attribute ai. Recursively, the same selection tree-building manner is carried out over every Tj, the place $1 < j < n$, till each subset is a member of a single class.

How to pick out the best attribute for every choice node when constructing the choice tree is the difficulty at hand. Gain Ratio Criterion is the criterion C4.5 selects. The necessary tenet of this criterion is to pick a attribute at every splitting stage that yields the most statistics obtain whilst minimizing the bias in want of exams with severa results via normalization.

A deci.sion tr.ee can be u.sed to categorize testing data that shares the sa.me properties as the training data after it has been constructed. The test is run on the same property of the testing case that the root node represents, beginning at the decision tree's root node. The branch whose condition is satisfied by the value of the tested attribute is chosen by the decision-making process. This branch directs the choice to a root node's offspring. Up until a leaf node is reached, the same procedure is repeated. The test case's specified class is connected to the leaf node.

3.12.2 Logistic Regression

Logistic regression evaluation is used to observe the relationship between a crew of impartial (explanatory) variables and a express based variable. The phrase "logistic regression" is used when the based variable solely has two values, such as zero and 1 or Yes and No. The time period multinomial logistic regression is usually used in instances the place the established variable has three or greater special values, such as Married, Single, Divorced, or Widowed.

The approach has a similar software in exercise to more than one regression, however the usage of a specific set of records for the based variable[78].

When it comes to evaluating categorical-response variables, discriminant analysis and logistic regression are competitors. Many statisticians believe that logistic regression, as opposed to discriminant analysis, is more adaptable and more suitable for modeling the majority of scenarios. This is so because, unlike discriminant analysis, logistic regression does not presuppose that the independent variables are regularly distributed.

A set of p independent variables $X_1, X_2, \dots, X_p$ are regressed on a categorical dependent variable Y having G (often $G = 2$) unique values in logistic regression. For instance, Y may represent the existence or absence of an illness, a patient's recovery from surgery, or their marital status. We frequently refer to these divisions by their sequential numbers because the names of these partitions are arbitrary. That is, Y will have the values 1, 2, ... G in the discussion that follows. In truth, NCSS permits Y to have both numeric and text values, although using integers makes the notation much clearer [78].

3.12.3 K-Nearest Neighbors

The K-Nearest-Neighbors (KNN) method makes no assumptions about the indispensable dataset due to the fact it is a non-parametric classification algorithm. It is properly recognised for being each effective and easy to use. It is an algorithm for supervised learning. To predict the classification of the unlabeled data, a top labeled education dataset with records factors divided into various lessons is supplied [79].

To determine which class the unlabeled data belongs to, classification uses a variety of characteristics. KNN is most frequently used as a classifier. Data is categorized using nearby or adjacent training examples in a certain location. This method is used because it computes quickly and is simple to use. It uses the Euclidean distance to locate its nearest neighbors when working with continuous data [79].

A new enter is given its K nearest neighbors, and the majority of the close by information is used to categorize the new input. Although this classifier is simple, the cost of "K" is critical for finding the unlabeled statistics.

There are many approaches to pick out the values for "K," however we can simply run the classifier with one-of-a-kind values a variety of instances to see which one yields the high-quality results. The computation value is a little larger due to the fact all computations are made whilst classifying the coaching statistics as a substitute than when it first seems in the dataset [79].

Because it solely shops the education information and memorizes the dataset when the dataset is being trained, it is a lazy getting to know method. On the coaching dataset, generalization is now not done. Therefore, when in the checking out stage, the complete education indispensable dataset is needed. KNN forecasts continuous values in regression. The common of this value's K closest neighbors' values makes up this fee.

3.12.4 Random Forest

Leo Breiman [80], who was motivated by earlier work by Amit and Geman [81], added random forests Breiman's bagging theory was expanded upon when Random Forests were developed as a counter to boosting [82], despite it not being immediately clear from the explanation in [80]. Both a continuous response, known as "regression," and a categorical response variable, known as "classification," can be employed with Random Forests. Similar to the dependent variables, the predictor variables might be categorical or continuous.

3.12.5 Gradient Boosting

Each training observation is first given the same weights through adaptive boosting (AdaBoost). It employs a number of weak models and gives data for which misclassification was noted larger weights. The accuracy of the misclassified observations is improved because it employs many weak models, incorporating the outcomes of the decision boundaries reached during multiple iterations, and as a result, the accuracy of the over

Gradient boosting (GB) [83], builds new models one at a time from a group of weak models with the hope that each new model will be able to reduce the loss function. The gradient descent method is used to calculate this loss function. Each new model fits the observations more closely when the loss function is used, increasing total accuracy. However, boosting must finally come to an end in order to prevent the model from being overfit. The terminating criteria may be a limit on how accurate the forecasts must be or a cap on the number of models.



4. SYSTEM IMPLEMENTATION

4.1 INTRODUCTION

This chapter is considered one of the main contributions to the thesis, as it contains the practical side of the study as it starts with the problem that the researcher studied, and the hardware and hardware components that were used in the project, also a detailed explanation about the data used and the method of generating it, then the selection of Decision Tree, Logistic Regression, k-nearest neighbors, Random Forest algorithms and how to design them. And training the neural network, then review the results of testing the designed network, and finally compare the results reached with some of the results of previous researchers in order to show the quality, accuracy and novelty of the proposed technology compared to the results reached above.

4.2 PROJECT TOOLS

Project tools divided into two main groups. The first one concern to the dataset collecting devices, and the second one concern to the machine learning python environment. The two groups specifications shown on the flowing sub paragraphs.

4.2.1 Hardware Description

i. ESP32.

The ESP32 is the heart of the project s a low-cost, low-power system on a chip microcontroller with built-in Wi-Fi and dual-mode Bluetooth. microcontroller board connects all of the sensors. The board is programmed with the source code in order to perform the project operations. In the memory on the chip provided in ESP32 the source code is stored, this can be considered as an interface between the programmer and the user that is why ESP32 is considered the heart of the project, in normal operating conditions the operating voltage range of ESP32 is 2.2 to 3.6V the ESP32 thing will supply 3.3V to the chip Figure4.3 depicts the ESP32 pin [84] .

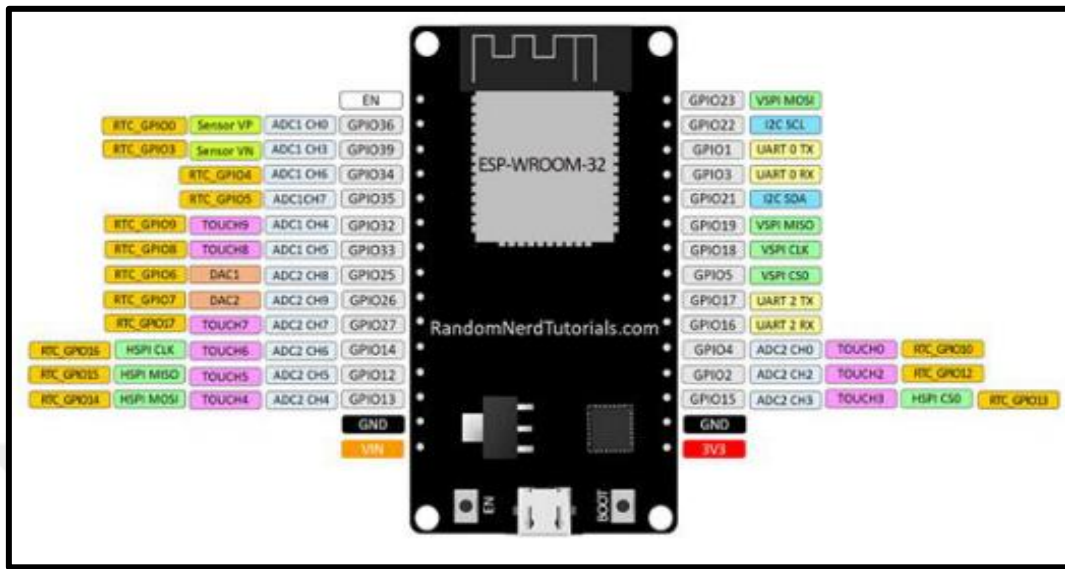


Figure 4.1: Pin description of ESP32 [85].

ii. ACS712 SENEOR

For accurate and affordable AC or DC current sensing in commercial, industrial, and communications applications, turn to the Allegro ACS712. The device package enables simple customer deployment. Motor control, load management, switched-mode power supplies, and overcurrent fault protection are examples of typical uses. The aspect consists of of an accurate, low-offset linear Hall sensor circuit with a copper conduction channel close to the floor of the die. The built-in Hall IC measures the magnetic discipline created through the utilized contemporary flowing thru this copper conduction line and converts it into a proportional voltage. The precision of the system is improved by means of the magnetic alerts that are shut to the Hall transducer.

The accurate, proportional voltage is delivered via the low-offset, chopper-stabilized BiCMOS Hall IC, which is tuned for accuracy after packaging. When a rising modern-day flows alongside the essential copper conduction path, which is the course utilised for contemporary sensing (from pins 1 and 2, to pins three and 4), the output of the system does certainly have a effective slope ($>V_{IOUT}(Q)$). This conductive channel normally has a resistance of 1.2 m, which consequences in minimal energy loss.

The system can live on beneath stipulations of up to a 5-fold overcurrent thanks to the thickness of the copper conductor. The sensor leads and the terminals of the conductive direction are electrically remoted (pins 5 thru 8). In functions requiring electrical isolation, the ACS712 contemporary sensor can now be usebarring the use of opto-isolators or different steeply-priced isolation techniques. The surface-mount SOIC8 packaging for the ACS712 is compact. It is possible to assemble printed circuit boards without lead (Pb) using the lead frame's 100% matte tin plating. Except for the flip-chip high-temperature Pb-based solder balls that are currently exempt from RoHS, the device is Pb-free internally. Prior to shipping from the factory, the device has been completely calibrated [84], (see Figure 4.2).



Figure 4.2: ACS712 Sensor [84].

iii. DHT11 SENSOR.

On the DHT11 Temperature & Humidity Sensor, a temperature and humidity sensor complicated outputs a calibrated digital signal. By combining temperature and humidity monitoring applied sciences with a precise digital sign accumulating methodology, it offers tremendous long-term balance and exquisite reliability. In order to grant gorgeous quality, fast response, interference resistance, and value effectiveness, this sensor connects to a high-performance 8-bit microcontroller and combines an NTC temperature size issue with a resistive-type humidity measuring element (See Figure 4.3).

Each DHT11 aspect is meticulously calibrated in a humidity calibration lab that is especially accurate. The calibration coefficients are used by way of the sensor's inner sign detecting technique and are saved as packages in the OTP memory. System integration is speedy and easy thanks to the single-wire serial interface. It is the greatest choice for a range of applications, which includes these that are the most demanding, due to the fact to its small size, low energy consumption, and up to 20-meter sign transmission. The issue is packaged with a single row of 4 pins. The capability to join is simple, and customers can request sure programs [84].

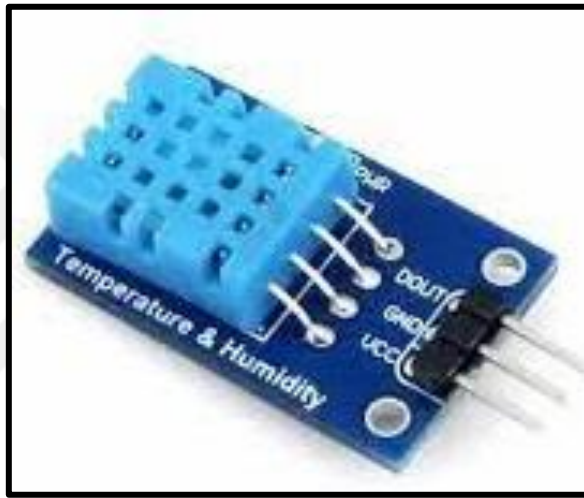


Figure 4.3: DHT11 Sensor [84].

4.2.2 Software Description

Python is an object-oriented programming language created by means of Guido Rossum in 1989. It is ideally designed for speedy prototyping of complicated applications. It includes interfaces to many running gadget calls and libraries. colab.reseaech.google) used to be a device for carrying out the learn about Collaboratory, or Colab for its abbreviation Colab for its abbreviation, is a product from Google Research that permits each person to write and execute Python code by a browser, and is specifically appropriate for computer learning, facts analysis, and education. It is a free cloud carrier hosted with the aid of (Google) to inspire Machine studying and synthetic Genius lookup [85].

Collaboratory permits the use of free GPUs and TPUs that can remedy all these problems. It consists of nearly all modules that can be used for records science analysis. These equipment consist of however are no longer restrained to Numpy, Scipy, Pandas, etc. It additionally consists of deep mastering frameworks, such as (Tensorflow), (Keras), (Pytorch) and different equipment and libraries. Table (4.1) indicates a evaluate of the most essential libraries that had been used in this find out about

Table 4.1: Python Libraries used in Project.

Library name	Description
Tensorflow	TensorFlow offers a set of procedures for Python model development and training, as well as quick deployment in the cloud, on-premises, in the browser, or on a device. This library has been used by Kersa, and Keras enables quick and simple neural network application prototyping. It supports seamless (GPU) networks, convolutional networks (CNN), redundant networks, as well as their clusters. [85].
Numpy	It is a Python package called "Scalar Python," which refers to a library with multidimensional array objects and a set of methods for manipulating matrices. [86].
Pandas	Using its potent data structures, the Pandas Python library offers a high performance data processing and analysis tool. The word "Panel Data," which refers to econometrics of multidimensional data, is where the name "Pandas" originates [86].
Sciklearn	It is Python's most practical and potent machine learning package. This library is mostly designed in Python and is based on NumPy, SciPy, and Matplotlib. It offers a variety of potent tools for statistical modeling and machine learning, including classification, regression, clustering, and dimension reduction via the consistency interface in Python [87]

Table 4.1: Python Libraries used in Project "tables continued".

Matplot library	As a cross-platform toolkit for drawing 2D diagrams from data in arrays, Matplotlib is one of the most well-known Python programs for data visualization. NumPy is used by Matplotlib, which is developed in Python. gives access to an object-oriented API that enables the incorporation of processors into software using Python GUI toolkits like PyQt and Wx Pythono Tkinter. This API can also be used with IPython, Jupyter, and web application servers [87].
Scipy	By giving the user high-level instructions and classes for data processing and visualization, the SciPy collection of mathematical methods and convenience functions greatly enhances the capabilities of an interactive Python session. By utilizing SciPy, an interactive Python session may compete with systems like MATLAB, IDL, Octave, R-Lab, and others as a data processing and system prototyping environment (SciLab)[88] [89].

4.3 DATASET

Power readings were obtained by reading the voltage and current of a certain number of household appliances based on Internet of Things devices and sensors. The data set collected during a period of three months of consecutive readings is provided. A number of devices were selected and shown in Table (4.2) and relevant information was obtained from packet capture files by extracting important features using tshark into comma separated values (.csv) files. The different devices were also categorized according to their device type and a category was assigned to the model training and testing process (See Appendix A).

The data collection mechanism can be expressed in the circuit shown in Figure (4.4) below, where ESP32 was used as a processor to upload the readings taken from the sensors to an Excel file in Google Drive, which was used as a cloud storage unit. The ACS712 part was used as a sensor to read the voltage and current, while the DHT11 sensor was used to read the ambient temperature.

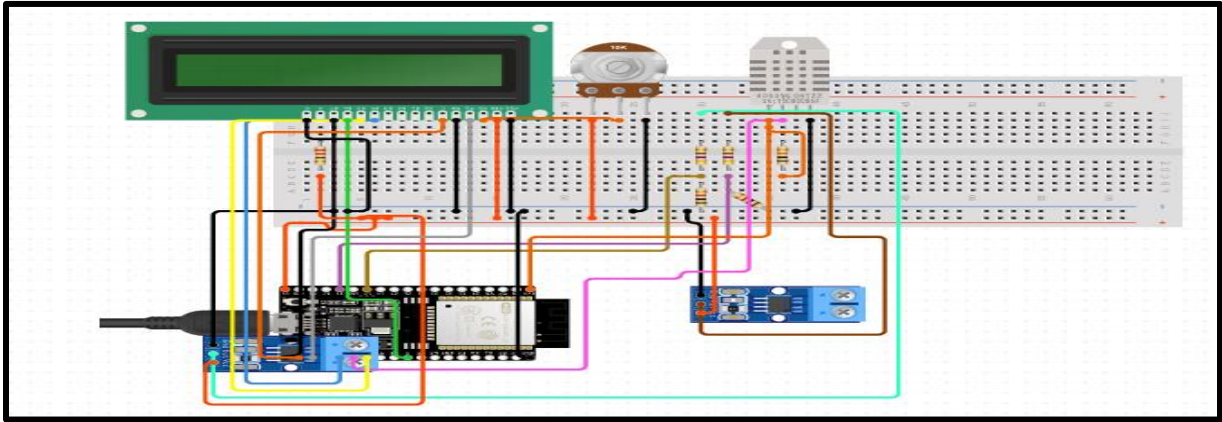


Figure 4.4: Dataset collection circuit Then the dataset was transferred from the drive and stored as csv file.

Table 4.2: Tested devices.

No	Device Name	Device Power (W)
1	3D printer	5.75
2	Air condition	113.25
3	Boiler	2506
4	Coffee Maker	3
5	Computer	3
6	Dish washing machine	3
7	Freezer	1
8	Refrigerator	60
9	Phone charger	2
10	Router	31

4.4 METHODOLOGE

The modern device uses electronics-based technology. The proposed solution uses electronic technologies to improve performance while lowering device costs. The proposed system consists of several components that help in detecting defects. Some components include database, control unit, home appliances, control unit, and fault maintenance unit.

The purpose of the control unit is to ensure that household appliances are working properly. When the program is activated, the console begins to monitor the data it receives from the console, which contains the most recent database of the device. From this, we conclude that the many algorithms can be used to detect faults in devices and store them in the fault maintenance unit with different accuracies. In addition, the console generates an alert message to the user when you expect a difference from the normal operation of the devices. The data is obtained daily from the console and stored in a file in the console's non-volatile memory, where it stores the data in the cloud using cloud computing technology.

The method calculates errors that have occurred or may occur using the stored data. This is the basic functional component of the entire system. The task of this controller is to support the proposed system protocol architecture. The proposed solution combines IoT and Internet technology to connect home appliances. The operation of the washing machine is tracked using the IoT sensor, including water usage, spin time, and running time of the washer, and data is sent to the control unit while monitoring the timing.

The console detects when a home appliance is turned on or off. When you turn on a device, it starts to work to determine if it is working fine. When an electrical appliance is turned on, the control unit begins to check if the appliance is working properly by providing the necessary amount of power to the running appliance. In addition, the control unit checks for problems with the device for its maintenance. The failure that occurs is stored in the controller because it has volatile memory, so the information is kept there until the controller requests it from the controller. It will use the basic algorithms which will have a probabilistic classification methodology. It is a very important unit.

The machine learning algorithm is intended to be data-driven future predictions, with the five algorithms mentioned in chapter three either classifying or collecting that data based on their similarities. This group data will be saved in a specific database where it will be used to predict future outcomes. In addition to this technique, the mechanism of support vectors will be used. By using this methodology to compare different data, the system will be able to predict the future.

This maintenance log has a distinct advantage that its server is free of errors. If this proposed system works by making the devices work to give data on the performance of those devices and then collect that data by IoT and then send it to the learning algorithm. Then the failure prediction method will be used and then the data will be kept in the fault history record. This prediction will be done with the help of the used algorithms that is similar to this The processor to deliver the data to the cloud and then there the input will be processed by the computer processor, just like analyzing the data and giving the user the final data.

The users will use the phone devices to search for information. When we say the end user, here we are referring to the phone, computer or any smart device that contains that The device has a database of devices, where this data helps in informing the device of any anomaly or difference in use.

4.5 PROJECT IMPLEMENTED AND RESULT

Fault detection System implemented in the Decision Tree, Logistic Regression, k-nearest neighbors, Random Forest, and Gradient Boosting algorithms on python colab platform. The model was tested against several batch sizes and epochs as well as classification criteria using the metrics presented in Table (4.3). The results shown in Tables and Figures below.

Table 4.3: The elements of the evaluation process (variables, definitions, and equations).

Variable	Definition	Equation
Accuracy	the percentage of accurately anticipated data from tests is easily determined by dividing all accurate forecasts by all predictions.	$Accuracy = \frac{Tp+Tn}{TP+TN+FP+FN}$
Precision	The proportion of outstanding instances among all anticipated ones from a specific class	$Precision = \frac{TP}{TP+FP}$
Recall	the ratio of the total number of occurrences to the proportion of instances that were supposed to be members of a class	$Recall = \frac{TP}{TP+FN}$
F1-Score	The phrase is used to describe a test's accuracy. The maximum F1-score is 1, which denotes outstanding recall and precision, while the lowest F1-score is 0.	$F1_Score = 2 \times \frac{precision \times recall}{Percison + recall}$

4.5.1 Result of Decision Tree classifier

In Decision Tree classifier the best getting accuracy equal to (96.67 %). The confusion matrix plot shown in figure (4.5) and the classification report shown in Table (4.4).

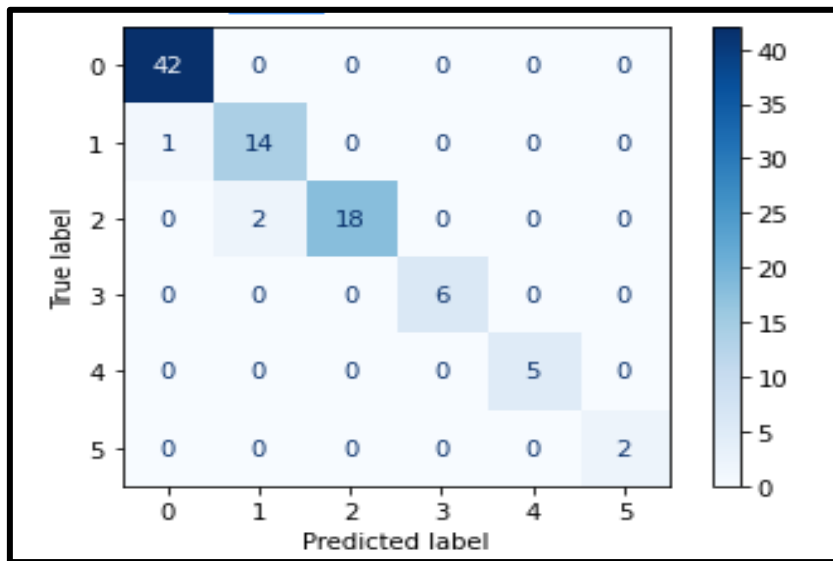


Figure 4.5: Decision Tree classifier confusion Matrix.

Table 4.4: Decision Tree classification report.

Try	Precision	Recall	F1-Score	Support
0	0.98	1	0.99	42
1	0.88	0.93	0.9	15
2	1	0.9	0.95	20
3	1	1	1	6
4	1	1	1	5
5	1	1	1	2
Accuracy = 0.97				
macro avg	0.98	0.97	0.97	90
weighted avg	0.97	0.97	0.97	90

4.5.2 Result of Logistic Regression classifier

In Logistic Regression classifier the best getting accuracy equal to (58.89 %). The confusion matrix plot shown in figure (4.6) and the classification report shown in Table (4.5).

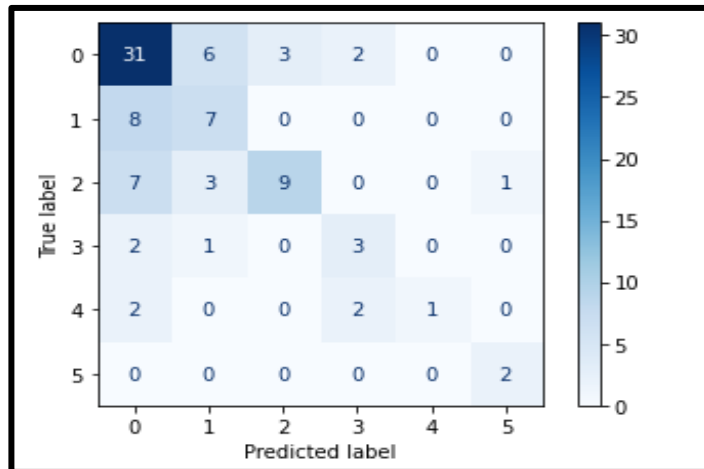


Figure 4.6: Logistic Regression classifier confusion Matrix.

Table 4.5: Logistic Regression classifier report.

Try	precision	recall	f1-score	support
0	0.62	0.74	0.67	42
1	0.41	0.47	0.44	15
2	0.75	0.45	0.56	20
3	0.43	0.5	0.46	6
4	1	0.2	0.33	5
5	0.67	1	0.8	2
accuracy = 0.59				
macro avg	0.65	0.56	0.54	90
weighted avg	0.62	0.59	0.58	90

4.5.3 Result of k-nearest neighbors classifier

In k-nearest neighbors classifier the best getting accuracy equal to (92.22%). The confusion matrix plot shown in figure (4.7) and the classification report shown in Table (4.6).

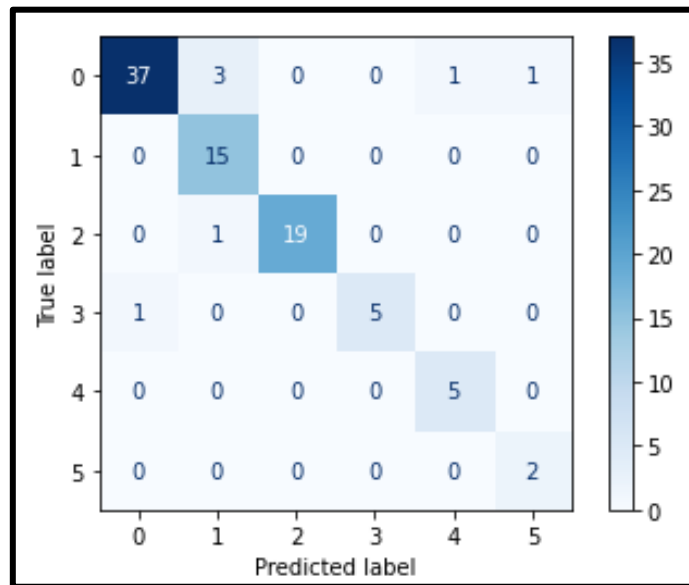


Figure 4.7: K-nearest neighbors classifier confusion Matrix.

Table 4.6: K-Nearest neighbors classifier report.

Try	Precision	Recall	F1-Score	Support
0	0.97	0.88	0.93	42
1	0.79	1	0.88	15
2	1	0.95	0.97	20
3	1	0.83	0.91	6
4	0.83	1	0.91	5
5	0.67	1	0.8	2
Accuracy = 0.92				
macro avg	0.88	0.94	0.9	90
weighted avg	0.94	0.92	0.92	90

4.5.4 Result of Random Forest

In Random Forest classifier the best getting accuracy equal to (97.78%). The confusion matrix plot shown in figure (4.8) and the classification report shown in Table (4.7).

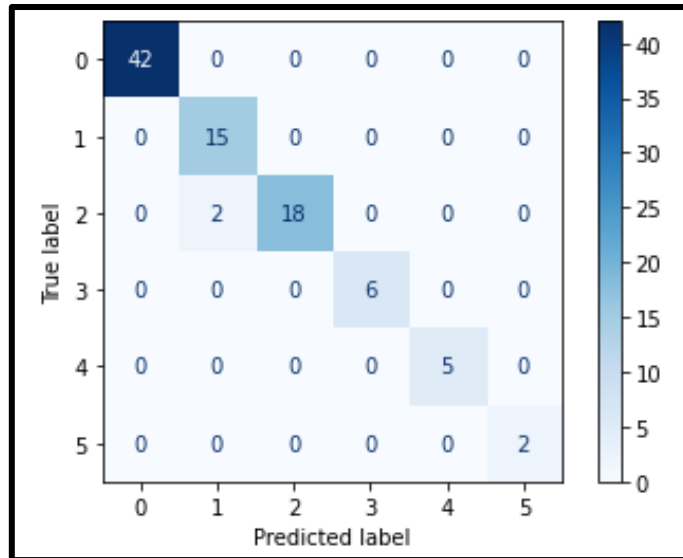


Figure 4.8: Random Forest classifier confusion Matrix.

Table 4.7: Random Forest classifier report

Try	Precision	Recall	F1-Score	Support
0	1	1	1	42
1	0.88	1	0.94	15
2	1	0.9	0.95	20
3	1	1	1	6
4	1	1	1	5
5	1	1	1	2
accuracy = 0.97				
macro avg	0.98	0.98	0.98	90
weighted avg	0.98	0.98	0.98	90

4.5.5 Result of Gradient Boosting

In Gradient Boosting classifier the best getting accuracy equal to (96.00%). The confusion matrix plot shown in figure (4.9) and the classification report shown in Table (4.8).

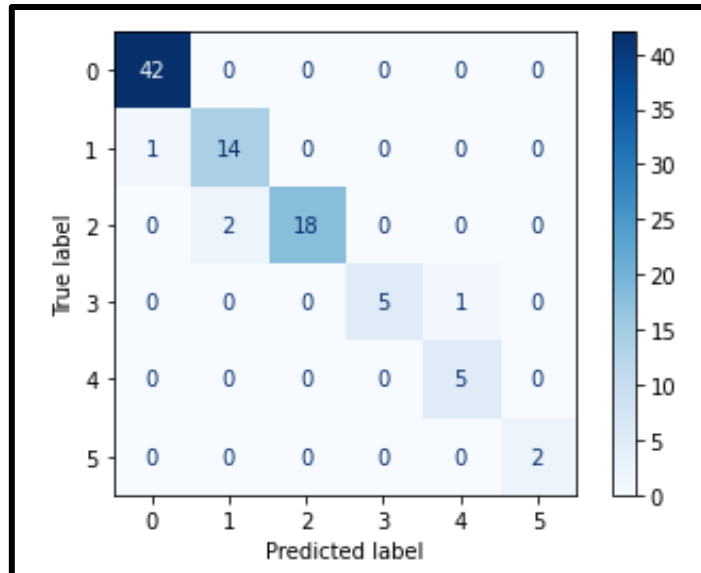


Figure 4.9: Gradient Boosting confusion Matrix.

Table 4.8: Gradient Boosting classifier report

Try	Precision	Recall	F1-Score	Support
0	0.98	1	0.99	42
1	0.88	0.93	0.9	15
2	1	0.9	0.95	20
3	1	0.83	0.91	6
4	0.83	1	0.91	5
5	1	1	1	2
accuracy = 0.96				
macro avg	0.95	0.94	0.94	90
weighted avg	0.96	0.96	0.96	90

4.6 RESULT SUMMARY

The results in previous paragraph summarized in Table (4.9), and after summary analysis the conclusion of the used algorithms utilize that Random Forest Classification algorithm has maximum accuracy.

Table 4.9: Results summary.

Algorithm	Accuracy	Precision	Recall	F1-Score	Support
Decision Tree Classifier	96.67%	0.98	0.97	0.97	90
Logistic Regression	58.89%	0.65	0.56	0.54	90
k-nearest neighbors	92.22%	0.88	0.94	0.9	90
Random Forest Classifier	97.78%	0.98	0.98	0.98	90
Gradient Boosting Classifier	96.00%	0.95	0.94	0.94	90

4.7 COMPARISON TO RELATIVE WORKS

After reviewing previous studies, a comparison was made between the algorithms used by the researchers and the algorithm used in this study, as well as comparing the accuracy that was obtained in all studies in order to distinguish this work from its predecessors. Table (4.10) illustrates this.

Table 4.10: Comparison to relative works.

Researcher	Testing Device	Data Acquisition Device	Dataset	Algorithm	Accuracy	Ref.
Abu-Samah et. al., 2015	Thermal treatment equipment	Dissolved gas concentrations data	Semiconductor manufacturer	BN	95.00%	[90]
Aydin & Guldamlasioglu,2017	Rotating equipment	Vibration	Vibration	DBN-FNN	85%	[91]
Mathew et. al 2017	Turbofan engine	NASA turbofan-engine dataset	NASA turbofan-engine dataset	kNN	-	[92]
Samuel et. al., 2017	Oil-immersed power transformer	-	Dissolved gas concentrations data	k-Means	94.59%	[93]
Deutsch & He,2018	Engine	NASA engine degradation simulation	NASA engine degradation simulation	LSTM	90.00%	[94]
Ma et. al., 2018	Aircraft engine	Engine degradation simulation	Engine degradation simulation	AE-LGR	84.19%	[95]
Kolokas et. al., 2018	Anode production of industrial equipment	Process sensor data	Process sensor data	DT	98.50%	[96]
Researcher	Testing Device	Data Acquisition Device	Dataset	Algorithm	Accuracy	Ref.

Table 4.10: Comparison to relative works " tables continued ".

Paolanti et. al., 2018	Cutting machine	PLCs and communication data	PLCs and communication data	RF	95.00%	[97]
Xiang et. al., 2018	Vending machine	Machine logs	Machine logs	GBM	80.00%	[98]
Huuhtanen & Jung, 2018	Photovoltaic panels	Electric power signals	Electric power signals	CNN	-	[99]
Sampaio et. al., 2019	Motor	Accelerometer	Vibration	ANN	31.30%	[100]
				RT	92.20%	
				RF	92.00%	
				SVM	69.60%	
Gohel et. al., 2020	Nuclear power plant	Temperature, power, current, speed	Temperature, power, current, speed	SVM	95.00%	[101]
Present Study	Different devices	Voltage, Current, Temperature	Voltage, Current, Temperature	DT	96.67%	-
				LR	58.89%	
				KNN	92.22%	
				RF	97.78%	
				Gradient Boosting	96.00%	

5. CONCLUSION AND FUTURE WORK

5.1 CONCLUSION

To provide robust customer management in smart homes, a framework that identifies faulty devices is required. Monitoring home appliances with a remote control enables the console to monitor and maintain as many home appliances as possible simultaneously. As a result, the proposed system aids in the detection of faults without the need for human intervention and saves a significant amount of time. Cloud computing and the Internet of Things are used to implement the proposed system. This is a multi-technology environment because the proposed system incorporates more than one technology. This method can be used to improve industrial devices in the future.

To predict electrical device failure, many algorithms were chosen. By simulating electrical appliance failures using the Internet of Things, the algorithm was tested on industrially generated IoT data sets. Because real-world IoT data typically has failures with respect to its size, synthetic IoT data was generated in three months to train and test the algorithms. A specific failure pattern is detected and injected into the data. The metrics used to evaluate the algorithm's prediction performance were accuracy and sensitivity. The algorithm was then tested on datasets that were generated using specified metrics.

The following key aspects were achieved by utilizing this service.

- i. Future device failure is predicted and avoided.
- ii. Based on the replacement rate, the manufacturer can predict the quality of their products.
- iii. An early increase in product production rate is predicted.
- iv. Estimate the product lifetime for a specific manufacturing plant.

5.2 FUTURE WORK

- i. Creating a free website available to all that predicts device failures by citizens.
- ii. Using other algorithms with more devices to make the project more comprehensive.
- iii. Designing an application for Android devices that performs the same purpose.
- iv. Designing a project that controls electrical appliances to determine the cost of the electricity wages calculation list and thus rationalize consumption.



REFERENCES

- [1] O. Aydin, S. Guldamlasioglu. “Using LSTM networks to predict engine condition on large scale data processing framework”. In: *2017 4th International Conference on Electrical and Electronic Engineering (ICEEE)*. 2017, pp. 281–285. doi: 10.1109/ICEEE2.2017.7935834 (cit. on pp. 26, 27).
- [2] B. L. Risteska Stojkoska, K. V. Trivodaliev. “A review of Internet of Things for smart home: Challenges and solutions”. In: *Journal of Cleaner Production* 140 (2017), pp. 1454–1464. issn: 0959 6526. doi: <https://doi.org/10.1016/j.jclepro.2016.10.006>. url: <https://www.sciencedirect.com/science/article/pii/S095965261631589X> (cit. on pp. 13, 17).
- [3] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee, B. Yin. “Smart Factory of Industry 4.0: Key Technologies, Application Case, and Challenges”. In: *IEEE Access* 6 (2018), pp. 6505–6519. doi: 10.1109/ACCESS.2017.2783682 (cit. on pp. 13, 17).
- [4] H. Guo, A. A. Ofori. “The Internet of Things in Extreme Environments Using Low Power Long Range Near Field Communication”. In: *IEEE Internet of Things Magazine* 4.1 (2021), pp. 34–38. doi: 10.1109/IOTM.0011.2000063 (cit. on p. 13).
- [5] M. Rafiuzzaman, J. Gascon Samson, K. Pattabiraman, S. Gopalakrishnan. “Failure Prediction in the Internet of Things Due to Memory Exhaustion”. In: *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*. SAC '19. Limassol, Cyprus: Association for Computing Machinery, 2019, pp. 292–301. isbn: 9781450359337. doi: 10.1145/3297280.3297311. url: <https://doi.org/10.1145/3297280.3297311> (cit. on pp. 13, 22, 23, 51).
- [6] N. Suga, K. Yano, J. Webber, Y. Hou, T. Higashimori, Y. Suzuki. “Prediction of QoS Outage Probability for Wireless Communication in Factory Environments”. In: *2019 International Conference on Internet of Things, Embedded Systems and Communications (IINTEC)*. Dec. 2019, pp. 124–129. doi: 10.1109/IINTEC48298.2019.9112098 (cit. on pp. 13, 24).

- [7] R. K. Mobley. *An introduction to predictive maintenance*. Elsevier, 2002 (cit. on pp. 13, 21).
- [8] M. Mohri, A. Rostamizadeh, A. Talwalkar. *Foundations of machine learning*. MIT press, 2018 (cit. on pp. 13, 19).
- [9] Z. M. Çınar, A. Abdussalam Nuhu, Q. Zeeshan, O. Korhan, M. Asmael, B. Safaei. "Machine Learning in Predictive Maintenance towards Sustainable Smart Manufacturing in Industry 4.0". In: *Sustainability* 12.19 (Oct. 2020), p. 8211. issn: 2071 1050. doi: 10.3390/su12198211. url: <http://dx.doi.org/10.3390/su12198211> (cit. on pp. 13, 26, 27).
- [10] T. Carvalho, F. Soares, R. Vita, R. Francisco, J. Basto, S. G. Soares Alcalá. "A systematic literature review of machine learning methods applied to predictive maintenance". In: *Computers & Industrial Engineering* 137 (Sept. 2019), p. 106024. doi: 10.1016/j.cie.2019.106024 (cit. on pp. 13, 26, 27).
- [11] A. C. Franco da Silva, P. Hirmer, J. Schneider, S. Ulusal, M. T. Frigo. "MBP: Not just an IoT Platform". In: *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. 2020, pp. 1–3. doi: 10.1109/PerComWorkshops48775.2020.9156156 (cit. on pp. 14, 18).
- [12] P. Hirmer, U. Breitenbücher, A. C. F. da Silva, K. Képes, B. Mitschang, M. Wieland. "Automating the Provisioning and Configuration of Devices in the Internet of Things." In: *CSIMQ* 9 (2016), pp. 28–43 (cit. on pp. 14, 18).
- [13] A.A.H. Mohamad, Y. S. Mezaal, S. F. Abdulkareem, "Computerized power transformer monitoring based on internet of things," *International Journal of Engineering & Technology* 7, no. 4, pp.2773 2778, 2018.
- [14] H. T. Salim, N. A. Jasim, "Design and Implementation of Smart City Applications Based on the Internet of Things," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 15, no. 13, pp. 4 15, 2021.

- [15] Wang, W.; Lee, K.; Murray, D. A global generic architecture for the future Internet of Things. *Serv. Oriented Comput. Appl.* 2017, 11, 329–344. [CrossRef]
- [16] Kua, J.; Nguyen, S.H.; Armitage, G.; Branch, P. Using Active Queue Management to Assist IoT Application Flows in Home Broadband Networks. *IEEE Internet Things J.* 2017, 4, 1399–1407. [CrossRef]
- [17] ZHILONG WANG, MIN ZHANG, DANSHI WANG, CHUANG SONG, MIN LIU, JIN LI, LIQI LOU, AND ZHUO LIU, “Failure prediction using machine learning and time series in optical network”, *Journal* 20C2A920201720 , (2017), <https://doi.org/10.1364/OE.25.018553>.
- [18] H. Li, X. Liu, Q. Mei. Predicting Smartphone Battery Life based on Comprehensive and Real time Usage Data. 2018. arXiv: 1801.04069 [cs.HC] (cit. on p. 24).
- [19] X. Fafoutis, A. Elsts, A. Vafeas, G. Oikonomou, R. Piechocki. “On Predicting the Battery Lifetime of IoT Devices: Experiences from the SPHERE Deployments”. In: *Proceedings of the 7th International Workshop on RealWorld Embedded Wireless Systems and Networks. RealWSN’18*. Shenzhen, China: Association for Computing Machinery, 2018, pp. 7–12. isbn: 9781450360487. doi: 10.1145/3277883.3277892. url: <https://doi.org/10.1145/3277883.3277892> (cit. on p. 24).
- [20] M. Rafiuzzaman, J. Gascon Samson, K. Pattabiraman, S. Gopalakrishnan. “Failure Prediction in the Internet of Things Due to Memory Exhaustion”. In: *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing. SAC ’19*. Limassol, Cyprus: Association for Computing Machinery, 2019, pp. 292–301. isbn: 9781450359337. doi: 10.1145/3297280.3297311. url: <https://doi.org/10.1145/3297280.3297311> (cit. on pp. 13, 22, 23, 51).
- [21] M. Soualhia, C. Fu, F. Khomh. “Infrastructure Fault Detection and Prediction in Edge Cloud Environments”. In: *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing. SEC ’19*. Arlington, Virginia: Association for Computing Machinery, 2019,

- pp. 222–235. isbn: 9781450367332. doi: 10.1145/3318216.3363305. url: <https://doi.org/10.1145/3318216.3363305> (cit. on p. 23).
- [22] K. Bhavana, V. Nekkanti, and N. Jayapandian, “Internet of things enabled device fault prediction system using machine learning,” in Proceedings of the International Conference on Inventive Computation Technologies, pp. 920–927, Springer, Coimbatore, India, August 2019.
 - [23] A. Xenakis, A. Karageorgos, E. Lallas, A. E. Chis, and H. V. Gonz’alez, “Towards distributed IoT/cloud based fault detection and maintenance in industrial automation,” *Procedia Computer Science*, vol. 151, pp. 683–690, 2019.
 - [24] A. Bhatnagar, S. Shukla, and N. Majumdar, “Machine learning techniques to reduce error in the internet of things,” in Proceedings of the 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), pp. 403–408, IEEE, Uttar Pradesh, India, January 2019.
 - [25] N. Suga, K. Yano, J. Webber, Y. Hou, T. Higashimori, Y. Suzuki. “Prediction of QoS Outage Probability for Wireless Communication in Factory Environments”. In: 2019 International Conference on Internet of Things, Embedded Systems and Communications (IINTEC). Dec. 2019, pp. 124–129. doi: 10.1109/IINTEC48298.2019.9112098 (cit. on pp. 13, 24).
 - [26] G. Lian, W. Chen, S. Huang. “Cloud Based Online Ageing Monitoring for IoT Devices”. In: IEEE Access 7 (2019), pp. 135964–135971. issn: 2169 3536. doi: 10.1109/ACCESS.2019.2941998 (cit. on p. 24).
 - [27] N. Pandey, O. P. Verma, A. Kumar. “A framework for usage pattern–based power optimization and battery lifetime prediction in smartphones”. In: Personal and Ubiquitous Computing (2019), pp. 1–16 (cit. on p. 24).
 - [28] Yyi Kai Teoh and Sukhpal Singh Gill and Ajith Kumar Parlikad, “IoT and Fog Computing based Predictive Maintenance Model for Effective Asset Management in

- Industry 4.0 using Machine Learning”, IEEE Internet of Things Journal, (2020). DOI:10.1109/JIOT.2021.3050441.
- [29] Power, Alexander and Kotonya, Gerald, “Providing Fault Tolerance via Complex Event Processing and Machine Learning for IoT Systems”, Association for Computing Machinery, 2019, <https://doi.org/10.1145/3365871.3365872>
- [30] [AMB20] T. Ara, P. M, M. Bali. “Fault Prediction in Wireless Sensor Networks using Soft Computing”. In: 2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE). Oct. 2020, pp. 532– 538. doi: 10.1109/ICSTCEE49637.2020.9277216 (cit. on p. 24).
- [31] A. Sour, A. S. Mohammed, M. P. Yousif, M. H. Malik, F. Safara, and M. Hosseinzadeh, “Formal verification of a hybrid machine learning based fault prediction model in Internet of things applications,” IEEE Access, vol. 8, Article ID 23863, 2020.
- [32] Bhavana, Kotte, Vinuthna Nekkanti, and N. Jayapandian. "Internet of things enabled device fault prediction system using machine learning." International Conference on Inventive Computation Technologies. Springer, Cham, 2019.
- [33] Zhang, Weishan, Qinghua Lu, Qiuyu Yu, Zhaotong Li, Yue Liu, Sin Kit Lo, Shiping Chen, Xiwei Xu and Liming Zhu. “Blockchain based Federated Learning for Failure Detection in Industrial IoT.” ArXiv abs/2009.02643 (2020): n. pag.
- [34] Bhavana, Kotte & Nekkanti, Vinuthna & N., Jayapandian. (2020). Internet of Things Enabled Device Fault Prediction System Using Machine Learning. 10.1007/978 3 030 33846 6_101.
- [35] Ouda, E., Maalouf, M. and Sleptchenko, A. Machine Learning and Optimization for Predictive Maintenance based on Predicting Failure in the Next Five Days. DOI: 10.5220/0010247401920199 In Proceedings of the 10th International Conference on Operations Research and Enterprise Systems (ICORES 2021), pages 192 199.

- [36] C. Wang, S. Feng, P. Liu, and X. Liu, "Design and development of collaborative office automation system based on hierarchical structure," in Proceedings of the E3S Web of Conferences, EDP Sciences, Dali, China, February 2021.
- [37] Mudita and D. Gupta, "A comprehensive study of recommender systems for the internet of things," Journal of Physics: Conference Series, IOP Publishing, vol. 1969, no. 1, 2021.
- [38] Imeri, Amil. *Prediction of failures of IoT devices by using machine learning*. MS thesis. 2021.
- [39] Wahid, A.; Breslin, J.G.; Intizar, M.A. Prediction of Machine Failure in Industry 4.0: A Hybrid CNN LSTM Framework. Appl. Sci. 2022, 12, 4221. <https://doi.org/10.3390/app12094221>.
- [40] Mihigo, I.N.; Zennaro, M.; Uwitonze, A.; Rwigema, J.; Rovai, M. On Device IoT Based Predictive Maintenance Analytics Model: Comparing TinyLSTM and TinyModel from Edge Impulse. Sensors 2022, 22, 5174. <https://doi.org/10.3390/s22145174>
- [41] L. Wallin, N. Jones, and S. Kleynhans, "How to Put an Implementable IoT Strategy in," no. July, 2015.
- [42] J. H. Choi, B. Yang, and C. W. Yu, "Artificial intelligence as an agent to transform research paradigms in building science and technology," *Indoor Built Environ.*, vol. 30, no. 8, pp. 1017–1020, 2021, doi: 10.1177/1420326X211017694.
- [43] [http://tblocks.com/internet of things](http://tblocks.com/internet%20of%20things)
- [44] Sethi, P., & Sarangi, S. R. (2017). Internet of things: architectures, protocols, and applications. *Journal of Electrical and Computer Engineering*, 2017.
- [45] Sethi, Sarangi, R. S. Internet of Things: Architectures, Protocols, and Applications. 2017. URL: <https://www.hindawi.com/journals/jece/2017/9324035/#B93> [Accessed 11 Dec 2018]

- [46] Khan R, Khan SU, Zaheer R, Khan S. Future internet: the internet of things architecture, possible applications and key challenges. In *Frontiers of Information Technology (FIT)*, 10th International Conference on 2012 Dec 17 (pp. 257 260). IEEE. 2012. URL: <https://pure.qub.ac.uk/portal/files/81384964/PID2566391.pdf> [Accessed 11 Dec 2018]
- [47] Gubbi J, Buyya R, Marusic S, Palaniswami M. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*. 2013 Sep;29(7):1645 60. URL: [https://www.scirp.org/\(S\(351jmbntvnsjt1aadkposzje\)\)/reference/ReferencesPapers.aspx?ReferenceID=1507740](https://www.scirp.org/(S(351jmbntvnsjt1aadkposzje))/reference/ReferencesPapers.aspx?ReferenceID=1507740) [Accessed 11 Dec]
- [48] Bonomi F, Milito R, Natarajan P, Zhu J. Fog computing: A platform for internet of things and analytics. In *Big data and internet of things: A roadmap for smart environments 2014* (pp. 169 186). Springer, Cham. URL: <http://www.sciepub.com/reference/122440> [Accessed 11 Dec 2018]
- [49] Stojmenovic I, Wen S. The fog computing paradigm: Scenarios and security issues. In *Computer Science and Information Systems (FedCSIS)*, 2014 Federated Conference on 2014 Sep 7 (pp. 1 8). IEEE. URL: <https://ieeexplore.ieee.org/document/6932989> [Accessed 11 Dec 2018]
- [50] Sethi, Sarangi, R. S. *Internet of Things: Architectures, Protocols, and Applications*. 2017. URL: <https://www.hindawi.com/journals/jece/2017/9324035/#B93> [Accessed 11 Dec 2018]
- [51] Salman Tin. A Summary of Network Traffic Monitoring and Analysis Techniques. 2018. URL: https://www.cse.wustl.edu/~jain/cse57015/ftp/iot_prot/#Wave [Accessed 11 Dec 2018]
- [52] HiveMQ. MQTT 101 How to get started with the lightweight IoT protocol. 2017. URL: <https://www.hivemq.com/blog/how-to-get-started-with-mqtt> [Accessed 11 Dec 2018]
- [53] Israel Koren and C. Mani Krishna. 2010. Preliminaries. In *Fault Tolerant Systems*. Morgan Kaufmann, Chapter 1, 400.

- [54] Algirdas Avizienis, Jean Claude Laprie, Brian Randell, and Carl Landwehr. 2004. Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing* 1, 1 (2004), 11–33. <https://doi.org/10.1109/TDSC.2004.2>
- [55] R. K. Mobley. *An introduction to predictive maintenance*. Elsevier, 2002 (cit. on pp. 13, 21).
- [56] Y. Ran, X. Zhou, P. Lin, Y. Wen, R. Deng. “A Survey of Predictive Maintenance: Systems, Purposes and Approaches”. In: *ArXiv abs/1912.07383* (2019) (cit. on pp. 21, 22, 27).
- [57] P. Zhao, M. Kurihara, J. Tanaka, T. Noda, S. Chikuma, T. Suzuki. “Advanced correlation based anomaly detection method for predictive maintenance”. In: *2017 IEEE International Conference on Prognostics and Health Management (ICPHM)*. 2017, pp. 78–83. doi: 10.1109/ICPHM.2017.7998309 (cit. on p. 22).
- [58] M. Rafiuzzaman, J. Gascon Samson, K. Pattabiraman, S. Gopalakrishnan. “Failure Prediction in the Internet of Things Due to Memory Exhaustion”. In: *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing. SAC '19*. Limassol, Cyprus: Association for Computing Machinery, 2019, pp. 292–301. isbn: 9781450359337. doi: 10.1145/3297280.3297311. url: <https://doi.org/10.1145/3297280.3297311> (cit. on pp. 13, 22, 23, 51).
- [59] L. Hu, L. Han, Z. Xu, T. Jiang, H. Qi. “A disk failure prediction method based on LSTM network due to its individual specificity”. In: *Procedia Computer Science* 176 (2020). Knowledge Based and Intelligent Information & Engineering Systems: Proceedings of the 24th International Conference KES2020, pp. 791–799. issn: 1877 0509. doi: <https://doi.org/10.1016/j.procs.2020.09.074>. url: <https://www.sciencedirect.com/science/article/pii/S1877050920319700> (cit. on p. 22).
- [60] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*. Elsevier Science, 2008.

- [61] I. H. Witten and E. Frank, Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, 2016.
- [62] J. Leskovec, A. Rajaraman, and J. D. Ullman, Mining of Massive Datasets. Cambridge university press, 2014.
- [63] M. M. Gorman, Database Management Systems: Understanding and Applying Database Technology. Elsevier Science, 2014
- [64] J. Fan, F. Han, and H. Liu, “Challenges of Big Data Analysis,” National Science Review, vol. 1, no. 2, pp. 293–314, 2014.
- [65] P. Zikopoulos, C. Eaton, et al., Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data. McGraw Hill Osborne Media, 2011
- [66] Michael A. Nielsen. Neural Networks and Deep Learning. Determination Press, 2015.
- [67] Antonino Laudani, Gabriele Maria Lozito, Francesco Riganti Fulginei, and Alessandro Salvini. On training efficiency and computational costs of a feed forward neural network: A review. Computational Intelligence and Neuroscience, 2015.
- [68] Song Han, Huizi Mao, and William J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. ICLR, 2016.
- [69] M. Mohri, A. Rostamizadeh, A. Talwalkar. Foundations of machine learning. MIT press, 2018 (cit. on pp. 13, 19).
- [70] S. Shalev Shwartz, S. Ben David. Understanding machine learning: From theory to algorithms. Cambridge university press, 2014 (cit. on p. 19).
- [71] D. W. Aha. Lazy learning. Springer Science & Business Media, 2013 (cit. on p. 19).
- [72] S. Shalev Shwartz, S. Ben David. Understanding machine learning: From theory to algorithms. Cambridge university press, 2014 (cit. on p. 19).

- [73] M. Seifeddine, A. Bradai, S. Bukhari, Q. Pham Tran Anh, O. Ben Ahmed, M. Atri. “A survey on machine learning in Internet of Things: Algorithms, strategies, and applications”. In: Internet of Things (Nov. 2020). doi: 10. 1016/j.iot.2020.100314 (cit. on p. 19).
- [74] M. A. Nielsen. Neural networks and deep learning. Vol. 25. Determination press San Francisco, CA, 2015 (cit. on p. 20).
- [75] A. Graves and J. Schmidhuber, “Offline handwriting recognition with multidimensional recurrent neural networks,” in Advances in neural information processing systems, 2009, pp. 545–552.
- [76] Jassas, M. (2022). *A framework for proactive fault tolerance in cloud IoT applications* (Doctoral dissertation)
- [77] Quinlan, J. R. (1993). C4.5, Programs for Machine Learning. Morgan Kaufmann San Mateo Ca, 1993.
- [78] A. Field, “Logistic regression Logistic regression Logistic regression,” Discovering Statistics Using SPSS. pp. 731–735, 2012.
- [79] K. Taunk, S. De, S. Verma, and A. Swetapadma, “A brief review of nearest neighbor algorithm for learning and classification,” 2019 Int. Conf. Intell. Comput. Control Syst. ICCS 2019, no. May 2019, pp. 1255–1260, 2019, doi: 10.1109/ICCS45141.2019.9065747.
- [80] Breiman, L.: Random Forests. Machine Learning 45 (1) pp. 5–32 (2001).
- [81] Amit, Y., Geman, D.: Shape quantization and recognition with randomized trees. Neural Computation 9(7) pp. 1545–1588 (1997).
- [82] Breiman, L.: Bagging Predictors. Machine Learning 24 (2) pp. 123–140 (2001).
- [83] Friedman, J.H. Stochastic gradient boosting. Comput. Stat. Data Anal. 2002, 38, 367–378.

- [84] Gay, Warren. "DHT11 sensor." *Advanced Raspberry Pi*. Apress, Berkeley, CA, 2018. 399–418.
- [85] P. Naik and G. Naik, *Conceptualizing Python in Google COLAB*, no. January. 2022.
- [86] Tutorials Point, "About the Tutorial Disclaimer & Copyright," *Organ. Behav.*, pp. 1–45, 2015.
- [87] E. Malinowski et al., "About the Tutorial Copyright & Disclaimer," *Data Vault 2.0*, no. January 1999, pp. 1–15, 2019, doi: 10.1007/978 3 322 94873 1.
- [88] C. Hill, "SciPy," *Learn. Sci. Program. with Python*, pp. 358–437, 2020, doi: 10.1017/9781108778039.009.
- [89] SciPy Community, "SciPy Reference Guide 0.7," p. 1229, 2013.
- [90] A. Abu Samah, M. Shahzad, E. Zamai, A. Said. "Failure Prediction Methodology for Improved Proactive Maintenance using Bayesian Approach". In: *IFAC PapersOnLine* 48.21 (2015). 9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes SAFEPROCESS 2015, pp. 844–851. issn: 2405 8963. doi: <https://doi.org/10.1016/j.ifacol.2015.09.632>. url: <http://www.sciencedirect.com/science/article/pii/S2405896315017619> (cit. on p. 27)
- [91] O. Aydin, S. Guldamlasioglu. "Using LSTM networks to predict engine condition on large scale data processing framework". In: *2017 4th International Conference on Electrical and Electronic Engineering (ICEEE)*. 2017, pp. 281–285. doi: 10.1109/ICEEE2.2017.7935834 (cit. on pp. 26, 27).
- [92] V. Mathew, T. Toby, V. Singh, B. M. Rao, M. G. Kumar. "Prediction of Remaining Useful Lifetime (RUL) of turbofan engine using machine learning". In: *2017 IEEE International Conference on Circuits and Systems (ICCS)*. 2017, pp. 306–311. doi: 10.1109/ICCS1.2017.8326010 (cit. on pp. 22, 27).
- [93] S. Eke, T. Aka Ngui, G. Clerc, I. Fofana. "Characterization of the operating periods of a power transformer by clustering the dissolved gas data". In: *2017 IEEE 11th International*

- Symposium on Diagnostics for Electrical Machines, Power Electronics and Drives (SDEMPED). 2017, pp. 298–303. doi: 10.1109/DEMPED.2017.8062371 (cit. on p. 27).
- [94] J. Deutsch, D. He. “Using Deep Learning Based Approach to Predict Remaining Useful Life of Rotating Components”. In: IEEE Transactions on Systems, Man, and Cybernetics: Systems 48.1 (2018), pp. 11–20. doi: 10.1109/TSMC.2017.2697842 (cit. on pp. 26, 27).
- [95] J. Ma, H. Su, W. l. Zhao, B. Liu. “Predicting the remaining useful life of an aircraft engine using a stacked sparse autoencoder with multilayer self learning”. In: Complexity 2018 (2018). doi: <https://doi.org/10.1155/2018/3813029> (cit. on pp. 26, 27).
- [96] N. Kolokas, T. Vafeiadis, D. Ioannidis, D. Tzovaras. “Forecasting faults of industrial equipment using machine learning classifiers”. In: 2018 Innovations in Intelligent Systems and Applications (INISTA). 2018, pp. 1–6. doi: 10.1109/INISTA.2018.8466309 (cit. on pp. 26, 27).
- [97] M. Paolanti, L. Romeo, A. Felicetti, A. Mancini, E. Frontoni, J. Loncarski. “Machine Learning approach for Predictive Maintenance in Industry 4.0”. In: 2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA). 2018, pp. 1–6. doi: 10.1109/MESA.2018.8449150 (cit. on pp. 26, 27)
- [98] S. Xiang, D. Huang, X. Li. “A Generalized Predictive Framework for Data Driven Prognostics and Diagnostics using Machine Logs”. In: TENCON 2018 2018 IEEE Region 10 Conference. 2018, pp. 0695–0700. doi: 10.1109/TENCON.2018.8650152 (cit. on pp. 26, 27).
- [99] T. Huuhtanen, A. Jung. “PREDICTIVE MAINTENANCE OF PHOTOVOLTAIC PANELS VIA DEEP LEARNING”. In: 2018 IEEE Data Science Workshop (DSW). 2018, pp. 66–70. doi: 10.1109/DSW.2018.8439898 (cit. on pp. 26, 27).
- [100] G. Scalabrini Sampaio, A. R. d. A. Vallim Filho, L. Santos da Silva, L. Augusto da Silva. “Prediction of Motor Failure Time Using An Artificial Neural Network”. In: Sensors

19.19 (Oct. 2019), p. 4342. issn: 1424 8220. doi: 10.3390/s19194342. url: <http://dx.doi.org/10.3390/s19194342> (cit. on pp. 26, 27).

- [101] H. A. Gohel, H. Upadhyay, L. Lagos, K. Cooper, A. Sanzetenea. “Predictive maintenance architecture development for nuclear infrastructure using machine learning”. In: Nuclear Engineering and Technology 52.7 (2020), pp. 1436–1442. issn: 1738 5733. doi:<https://doi.org/10.1016/j.net.2019.12.029>. url: <http://www.sciencedirect.com/science/article/pii/S1738573319306783> (cit. on pp. 26, 27).



APPENDIX A

Data set Sample

device	manufacture	ratedVoltage	max power	max current	Voltage(V)	Current(A)	Power(W)	Energy(KWPH)	Frequency(Hz)	PF	Temp (C)	status
TV	LG 50um7300aue	220	111.32	0.55	208.4	0.51	98.5	0.074	50.4	0.92	28.75	normal
TV	LG 50um7300aue	220	111.32	0.55	208.2	0.51	98.5	0.074	50.4	0.92	29	normal
TV	LG 50um7300aue	220	111.32	0.55	206.9	0.52	98.6	0.075	50.4	0.92	28.5	normal
TV	LG 50um7300aue	220	111.32	0.55	209.1	0.51	98.7	0.075	50.3	0.92	29	normal
TV	LG 50um7300aue	220	111.32	0.55	206.6	0.52	98.5	0.075	50.4	0.92	28.5	normal
TV	LG 50um7300aue	220	111.32	0.55	207.4	0.51	98	0.076	50.4	0.92	29.25	normal
TV	LG 50um7300aue	220	111.32	0.55	206.8	0.51	98.4	0.076	50.3	0.92	29.25	normal
TV	LG 50um7300aue	220	111.32	0.55	205.4	0.52	98	0.076	50.3	0.92	28.5	normal
TV	LG 50um7300aue	220	111.32	0.55	202	0.39	71.2	0.087	50.2	0.9	29	normal
TV	LG 50um7300aue	220	111.32	0.55	204	0.39	71.1	0.087	50.2	0.9	29.5	normal
TV	LG 50um7300aue	220	111.32	0.55	202.1	0.39	70.9	0.088	50.2	0.9	29.25	normal
TV	LG 50um7300aue	220	111.32	0.55	203	0.32	55.4	0.088	50.2	0.86	29.5	normal
TV	LG 50um7300aue	220	111.32	0.55	201.6	0.39	71	0.088	50.2	0.9	29.25	normal
TV	FHD LED 42	220	66	0.5	197.9	0.39	47.5	0.09	50.1	0.61	28	normal
TV	FHD LED 42	220	66	0.5	198.6	0.39	47.5	0.09	50.1	0.61	27.5	normal
TV	FHD LED 42	220	66	0.5	199.7	0.39	47.7	0.09	50.1	0.61	28.25	normal
TV	FHD LED 42	220	66	0.5	197.5	0.4	47.6	0.09	50.1	0.61	27.75	normal
TV	FHD LED 42	220	66	0.5	198.6	0.39	47.6	0.09	50.1	0.61	28.25	normal
TV	FHD LED 42	220	66	0.5	198.2	0.39	47.2	0.09	50.2	0.61	28.25	normal
TV	FHD LED 42	220	66	0.5	197.7	0.4	47.4	0.091	50.1	0.61	27.75	normal
TV	FHD LED 42	220	66	0.5	198.2	0.4	47.4	0.091	50.2	0.6	27.75	normal

TV	FHD LED 42	220	66	0.5	200	0.39	47.3	0.091	50.1	0.6	28	normal
TV	FHD LED 42	220	66	0.5	198.9	0.4	47.6	0.091	50.1	0.6	28	normal
TV	FHD LED 42	220	66	0.5	196.6	0.4	47.3	0.091	50.1	0.6 1	28	normal
TV	FHD LED 42	220	66	0.5	198.6	0.39	47.5	0.091	50.1	0.6 1	27.75	normal
TV	FHD LED 42	220	66	0.5	199.5	0.39	47.4	0.091	50.1	0.6 1	27.75	normal
TV	FHD LED 42	220	66	0.5	199.8	0.39	47.4	0.091	50.2	0.6	27	normal
TV	FHD LED 42	220	66	0.5	201.1	0.38	47.2	0.092	50.1	0.6 1	27.75	normal



Algorithm Code sample

66