

# **SMART HOME MONITORING SYSTEM USING MACHINE LEARNING AND IOT**

Submitted in partial fulfillment of the requirements for the award of Bachelor  
of Engineering Degree in Electronics and Communication Engineering

by

**D.PRADEEP (40130040)**

**M.VISHNU VARDHAN (40130128)**



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING  
SCHOOL OF ELECTRICAL AND ELECTRONICS**

## **SATHYABAMA**

**INSTITUTE OF SCIENCE AND TECHNOLOGY  
(DEEMED TO BE UNIVERSITY)**

**CATEGORY -1 UNIVERSITY BY UGC**

**Accredited with Grade "A++" by NAAC | 12B Status by UGC | Approved by AICTE**

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600119**

**APRIL - 2024**



# **SATHYABAMA**

INSTITUTE OF SCIENCE AND TECHNOLOGY  
(DEEMED TO BE UNIVERSITY)

Accredited "A++" Grade by NAAC | 12B Status by UGC | Approved by AICTE  
[www.sathyabama.ac.in](http://www.sathyabama.ac.in)

## **DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

### **BONAFIDE CERTIFICATE**

This is to certify that this Project Report is the bonafide work of M.VISHNU VARDHAN (40130128), D.PRADEEP (40130040) who carried out the project entitled "**SMART HOME MONITORING SYSTEM USING MACHINE LEARNING AND IOT**" under our supervision from November 2023 to April 2024.

**Internal Guide**

Dr. R. PANDIAN, ME., Ph.D.  
Professor

**Head of the Department**

Dr. T. RAVI, M.E., Ph.D.  
Professor

---

Submitted for Viva voce Examination held on 18/04/2024

Internal Examiner

External Examiner

## DECLARATION

I, **M.VISHNU VARDHAN (40130128)** hereby declare that the Project Report entitled "**SMART HOME MONITORING SYSTEM USING MACHINE LEARNING AND IOT** " done by us under the guidance of Dr. R. PANDIAN, M.E,Ph.D,Professor, Dept. of Electronics and Communication Engineering, SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY, CHENNAI, is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

DATE: 18/04/24

PLACE: Chennai

### SIGNATURE OF THE CANDIDATES

1. M.vishnu vardhan
2. D.Pradeep

## ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to the **Board of Management of SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. N. M. NANDHITHA, M.E., Ph.D., Dean, School of Electrical and Electronics Engineering** and **Dr. T. RAVI, M.E., Ph.D., Head of the Department, Department of Electronics and Communication Engineering** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.R.PANDIAN, M.E., Ph.D., Professor, Dept. of Electronics and Communication Engineering** for his valuable guidance, suggestions and constant encouragement for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Electronics and Communication Engineering who were helpful in many ways for the completion of the project.

## **ABSTRACT**

Electrical energy is that the lifeblood of modern societies. Within the past decades, the world's energy consumption and associated CO<sub>2</sub> emissions exaggerated quickly due to the increases in population and luxury demands of people. Building electrical energy consumption prediction is essential for electrical energy planning, management, and conservation. This project work aims to develop a machine learning model, method, architecture or appliance to reduce building energy use and emissions using a smart sensor for residential or commercial buildings. An experienced operator can do a good job of adjusting set points and schedule. But no matter how good they are, a human's ability is restricted by the amount of knowledge he or she can process. Significant opportunities exist to take advantage of external data sources including real-time occupancy sensor networks, changing space schedules, weather forecasts, grid carbon intensity, and other environmental conditions that could help us better predict space set-points and schedules 24x7. The sensor senses CO<sub>2</sub> reading, ambient light, door state sensing etc. These can be used to accurately estimate the number of occupants in each room using machine learning techniques and this technique can be used to predict future occupancy and also if the CO<sub>2</sub> level reaches the threshold value the mail will be sent to the user.

## TABLE OF CONTENTS

| CHAPTER NO | TITLE                                         | PAGE NO |
|------------|-----------------------------------------------|---------|
|            | <b>ABSTRACT</b>                               | v       |
|            | <b>LIST OF FIGURES</b>                        | viii    |
| <b>1</b>   | <b>INTRODUCTION</b>                           | 1       |
|            | 1.1 Introduction                              | 1       |
|            | 1.2 Internet of Things                        | 3       |
|            | 1.3 Machine Learning                          | 6       |
| <b>2</b>   | <b>LITERATURE SURVEY</b>                      | 11      |
| <b>3</b>   | <b>AIM AND SCOPE OF PRESENT INVESTIGATION</b> | 14      |
|            | 3.1 Aim                                       | 14      |
|            | 3.2 Scope of the project                      | 15      |
| <b>4</b>   | <b>EXPERIMENTAL METHODS</b>                   | 17      |
|            | 4.1 Existing System                           | 17      |
|            | 4.2 Proposed System                           | 17      |
|            | 4.3 Proposed Block Diagram                    | 18      |
|            | 4.4 Hardware Requirement                      | 22      |
|            | 4.4.1 Arduino UNO                             | 22      |
|            | 4.4.2 MQ2 Sensor                              | 26      |
|            | 4.4.3 LDR Sensor                              | 27      |
|            | 4.4.4 PIR Sensor                              | 29      |
|            | 4.4.5 DHT11 Sensor                            | 30      |
|            | 4.4.6 LCD Display                             | 32      |
|            | 4.4.7 DC Motor                                | 33      |
|            | 4.4.8 Power Supply                            | 35      |
|            | 4.4.9 Transformer                             | 36      |
|            | 4.5 Software Requirement                      | 38      |
|            | 4.5.1 Arduino ide                             | 38      |
|            | 4.5.2 Visual Studio Code                      | 38      |

|          |                                      |    |
|----------|--------------------------------------|----|
|          | 4.6 Functionality of Proposed Work   | 39 |
|          | 4.6.1 Standards                      | 39 |
|          | 4.6.2 Constraints                    | 40 |
|          | 4.6.3 Tradeoffs                      | 41 |
| <b>5</b> | <b>RESULT AND DISCUSSIONS</b>        | 43 |
|          | 5.1 Performance Result               | 43 |
|          | 5.2 Performance of Mail Alert System | 45 |
| <b>6</b> | <b>CONCLUSION AND FUTURE SCOPE</b>   | 50 |
|          | 6.1 Conclusion                       | 50 |
|          | 6.2 Future Scope                     | 50 |
|          | <b>REFERENCES</b>                    | 51 |

## LIST OF FIGURES

| FIGURE NO | FIGURES NAME                                | PAGE NO. |
|-----------|---------------------------------------------|----------|
| 1.1       | IOT Architecture                            | 4        |
| 1.2       | Classification of Machine Learning          | 7        |
| 4.1       | Proposed Block Diagram                      | 18       |
| 4.2       | Block Diagram of Machine Learning           | 20       |
| 4.3       | Arduino Board                               | 23       |
| 4.4       | Arduino Uno Pin Configuration               | 25       |
| 4.5       | MQ2 Sensor                                  | 27       |
| 4.6       | LDR Sensor                                  | 28       |
| 4.7       | PIR Sensor                                  | 30       |
| 4.8       | DHT11 Sensor                                | 31       |
| 4.9       | 16x2 LCD Display                            | 33       |
| 4.10      | DC Motor                                    | 35       |
| 4.11      | Generic Block Diagram of Power Supply Unit  | 35       |
| 4.12      | Transformer                                 | 37       |
| 5.1       | Working Model of Proposed Work              | 44       |
| 5.2       | Mail Alert for Smart Home CO2 Level         | 45       |
| 5.3       | CO2 Level Graph using Dataset               | 46       |
| 5.4       | Humidity and LDR Levels using Dataset       | 47       |
| 5.5       | PIR and Overall Sensor Levels using Dataset | 47       |
| 5.6       | KNN Implementation                          | 48       |
| 5.7       | Naive Bayes Implementation                  | 48       |

# **CHAPTER-1**

## **INTRODUCTION**

### **1.1 INTRODUCTION**

Excessive domestic electrical energy usage is an impediment towards energy potency. Developing countries are expected to witness an associate new rise in domestic electricity in the forthcoming decades. A large quantity of research has been directed towards behavioral change for energy efficiency. Emission of greenhouse gases including carbon dioxide in higher layers of the atmosphere are referred as the main cause of global warming phenomena. The plan to decrease the quantity of green- house gases wants vital alteration in human behavior in energy consumption, producing of more environmental friendly products and distinguishing and mitigating the causes of these undesirable gases. In traditional buildings, households are responsible for continuously observance and controlling the installed Heating, Ventilation, and Air Conditioning (HVAC) system. Unnecessary energy consumption might occur due to,as an example, forgetting devices turned on, that overwhelms users because of the necessity to tune the devices manually. So,we have determined to build a smart building,which are automating this method by automatically tuning HVAC systems consistent with user preferences so as to boost user satisfaction and optimize energy consumption.

In recent years, the convergence of Internet of Things (IoT) technology and advancements in machine learning has revolutionized various aspects of daily life. One area profoundly impacted by this synergy is the realm of smart home systems. Smart home monitoring systems have emerged as a cornerstone application, offering homeowners unprecedented levels of convenience, security, and efficiency. By integrating IoT devices with sophisticated machine learning algorithms, these systems can intelligently monitor and manage various aspects of the home environment in real-time. the integration of IoT sensors enables smart home systems to collect vast amounts of data on environmental conditions and user preferences. Machine learning algorithms analyze this data to provide personalized recommendations and automate routine tasks, such as adjusting thermostat settings or optimizing energy usage. Machine learning in smart home monitoring systems is its ability to adapt and personalize based on individual preferences and habits.

The primary objective of smart home monitoring systems is to enhance the overall quality of life for occupants while simultaneously optimizing energy usage and ensuring security. Traditional home monitoring systems often rely on static rules or predefined schedules, which may not adequately adapt to dynamic environmental conditions or user preferences. However, by harnessing the power of IoT sensors and actuators coupled with machine learning techniques, these systems can dynamically respond to changing conditions, learn from user behavior, and make intelligent decisions autonomously.

IoT devices such as smart thermostats, motion sensors, door/window sensors, surveillance cameras, and smart appliances serve as the sensory network within a smart home ecosystem. These devices continuously collect a wealth of data related to temperature, occupancy, energy consumption, and security events. By leveraging this rich stream of sensory data, machine learning algorithms can analyze patterns, detect anomalies, and extract meaningful insights to inform decision-making processes.

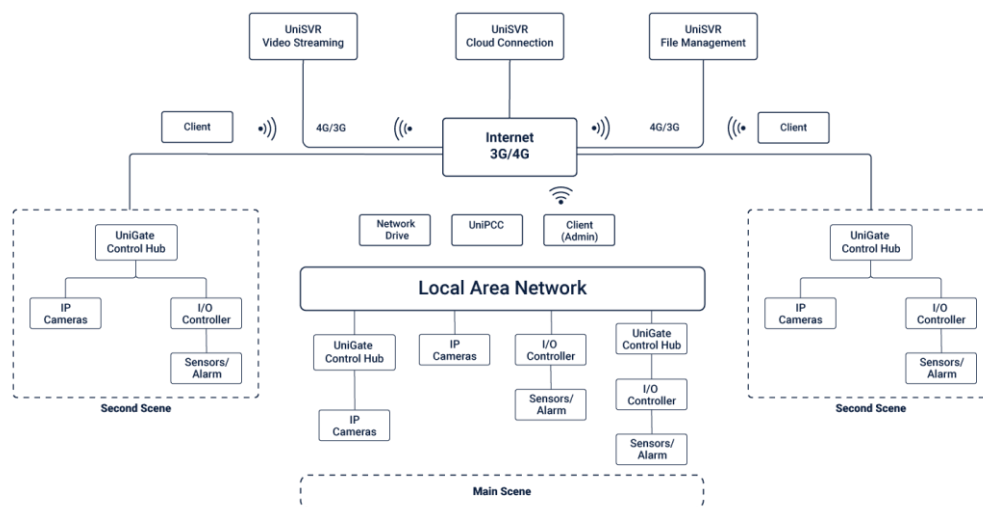
One of the key advantages of employing machine learning in smart home monitoring systems is its ability to adapt and personalize based on individual preferences and habits. For example, machine learning algorithms can learn the occupants' daily routines, anticipate their needs, and adjust environmental settings accordingly. This personalized approach not only maximizes comfort but also minimizes energy wastage by optimizing heating, cooling, and lighting based on occupancy patterns.

Furthermore, machine learning algorithms play a crucial role in enhancing the security capabilities of smart home monitoring systems. By continuously analyzing sensor data and user interactions, these algorithms can detect unusual behaviors or potential security threats, such as unauthorized access attempts or anomalous activities within the home environment. This proactive approach to security empowers homeowners with real-time alerts and preventive measures, thereby enhancing overall safety and peace of mind. smart home monitoring systems offer remote access and control via smartphone applications, allowing homeowners to manage their homes from anywhere in the world.

## **1.2 INTERNET OF THINGS**

The Internet of Things (IoT) represents a transformative paradigm in which everyday objects are connected to the internet, enabling them to collect, exchange, and analyze data. This interconnected network of devices, sensors, and systems has the potential to revolutionize numerous industries and domains, from healthcare and transportation to agriculture and manufacturing. At its core, IoT enables seamless communication between physical objects and digital systems, creating opportunities for automation, optimization, and innovation. By embedding sensors and connectivity into devices, IoT enables real-time monitoring and control, predictive maintenance, and data-driven decision-making.

In the realm of smart homes, IoT devices such as thermostats, cameras, and appliances can be interconnected to create intelligent environments that enhance comfort, convenience, and energy efficiency. In healthcare, wearable sensors and medical devices can monitor patient vital signs remotely, enabling early detection of health issues and personalized care. In industrial settings, IoT enables the concept of Industry 4.0, where interconnected machines, sensors, and systems enable smart factories that optimize production processes, minimize downtime, and improve product quality. In agriculture, IoT sensors can monitor soil moisture levels, weather conditions, and crop health, enabling precision farming techniques that enhance yields while conserving resources. However, with the proliferation of IoT devices comes challenges related to security, privacy, interoperability, and data management. Ensuring the security and privacy of IoT systems is paramount to protect against cyber threats and unauthorized access to sensitive data. Interoperability standards and protocols are essential to enable seamless communication and integration between diverse IoT devices and platforms. Additionally, effective data management strategies are necessary to handle the vast amounts of data generated by IoT devices, ensuring data quality, integrity, and accessibility. IoT also has the potential to enhance urban living by creating smart cities equipped with connected infrastructure, such as intelligent transportation systems, energy-efficient buildings, and waste management solutions. These advancements can improve public safety, reduce traffic congestion, and mitigate environmental impact.



**Fig 1.1 IOT Architecture**

The above Fig 1.1 shows the Internet of Things architecture diagram. It contains the Arduino board block and it is connected to the various sensors and modules like RF module and it is connected with Access Point which is wifi or bluetooth and there is a driver circuit, which is used to connect to the various home appliances and the data from the home appliances and other appliances are connected to the IoT server. The IoT server is connected to the cloud. The data from the server is sent to the cloud and data is stored in cloud. Again, the data in cloud is sent to the server. Also that cloud storage is connected to the PC terminal.

**Enhanced Efficiency and Automation:** IoT enables automation of various tasks and processes by connecting devices and systems. This automation leads to increased efficiency, reduced human intervention, and optimized resource utilization. For example, in manufacturing, IoT-enabled sensors and actuators can automate production processes, monitor equipment health, and schedule maintenance tasks, leading to higher productivity and cost savings.

**Data-Driven Insights and Decision Making:** IoT generates vast amounts of data from connected devices and sensors. By analyzing this data using advanced analytics and machine learning algorithms, organizations can gain valuable insights into operations, customer behavior, and market trends. These insights enable data-driven decision-making, allowing businesses to identify opportunities, mitigate risks, and improve overall performance.

**Improved Safety and Security:** IoT plays a crucial role in enhancing safety and security across various domains. In smart cities, IoT-enabled surveillance cameras, traffic sensors, and emergency response systems help monitor and manage public safety. In healthcare, IoT devices such as wearable monitors and remote patient monitoring systems enable timely intervention and emergency response. Additionally, IoT security solutions help protect connected devices and networks from cyber threats and unauthorized access.

**Sustainable Practices and Environmental Monitoring:** IoT contributes to sustainability efforts by enabling smart resource management and environmental monitoring. For instance, in smart buildings, IoT sensors can optimize energy usage, monitor air quality, and manage waste, leading to reduced environmental impact and cost savings. In agriculture, IoT-enabled precision farming techniques minimize water usage, optimize fertilizer application, and improve crop yields, promoting sustainable agricultural practices.

**Enhanced User Experience and Personalization:** IoT enables personalized experiences and services by collecting and analyzing data about user preferences, behavior, and context. For example, IoT devices in retail stores can track customer movements, analyze purchase history, and deliver personalized recommendations and promotions in real-time. Similarly, in smart homes, IoT-enabled devices can adjust settings and preferences based on user behavior and preferences, enhancing comfort and convenience.

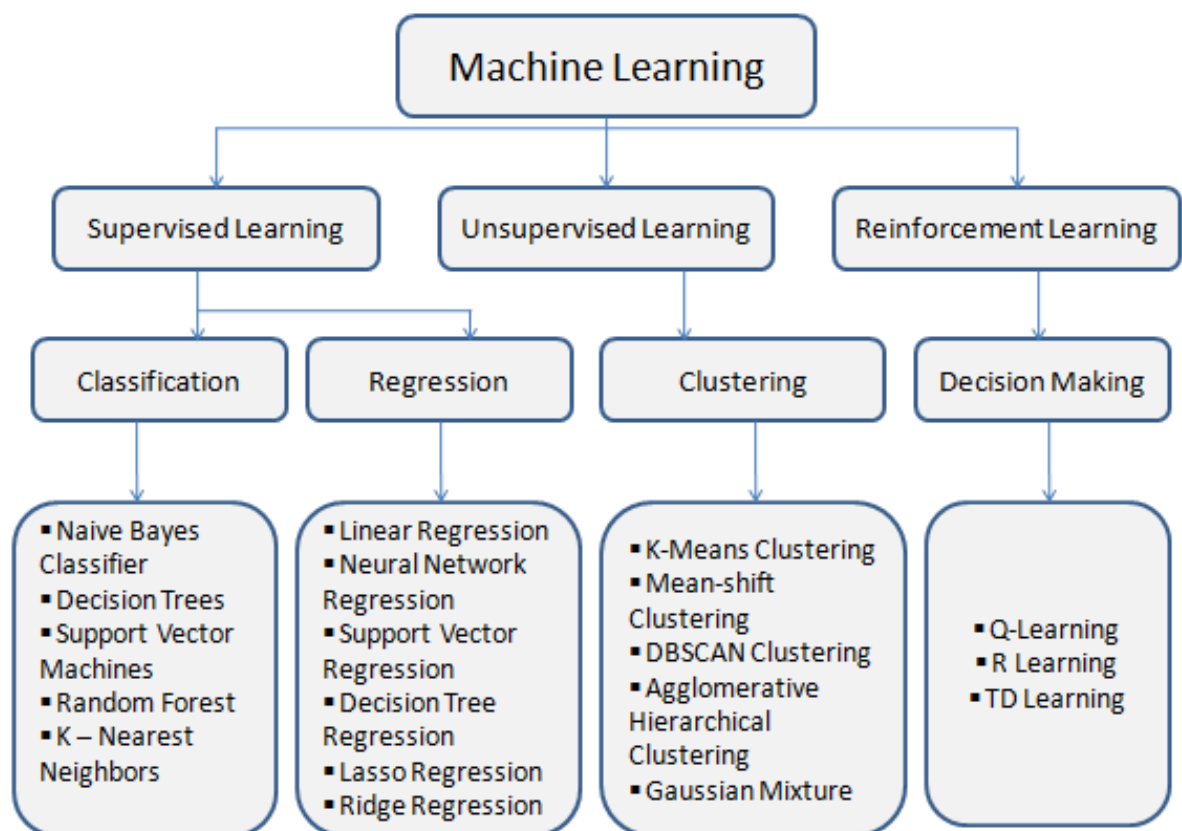
IoT has the potential to transform industries, improve quality of life, and drive innovation in countless ways. By harnessing the power of connectivity, data, and intelligence, IoT promises to reshape the way we interact with the world around us, creating smarter, more efficient, and more sustainable environments for generations to come. the potential of the Internet of Things (IoT) is vast and multifaceted, touching nearly every aspect of our lives and industries. One of the most exciting aspects of IoT is its ability to revolutionize industries by optimizing processes, increasing efficiency, and reducing costs. For example, in manufacturing, IoT-enabled sensors can monitor equipment performance in real-time, predict maintenance needs, and minimize downtime, leading to increased productivity and reduced operational costs.

### **1.3 MACHINE LEARNING**

Machine learning is a branch of artificial intelligence (AI) that focuses on the development of algorithms and statistical models that enable computers to learn from and make predictions or decisions based on data, without being explicitly programmed for specific tasks. Machine learning algorithms analyze large datasets, identify patterns, and make informed predictions or decisions without human intervention. There are several types of machine learning algorithms, including supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. In supervised learning, the algorithm learns from labeled data, where each input is associated with a corresponding output label. The algorithm learns to map inputs to outputs, making predictions or classifications based on new, unseen data. In unsupervised learning, the algorithm learns from unlabeled data, identifying patterns and structures in the data without explicit guidance. Semi-supervised learning combines elements of both supervised and unsupervised learning, using a small amount of labeled data and a larger amount of unlabeled data. Reinforcement learning involves training an agent to interact with an environment and learn from feedback in the form of rewards or penalties.

Machine learning is widely used in various applications and industries, including healthcare, finance, marketing, transportation, and manufacturing. In healthcare, machine learning algorithms can analyze medical images, predict patient outcomes, and assist in diagnosis and treatment planning. In finance, machine learning is used for fraud detection, risk assessment, algorithmic trading, and personalized financial services. In marketing, machine learning algorithms analyze customer behavior, segment customers, and personalize marketing campaigns. In transportation, machine learning enables autonomous vehicles, traffic management systems, and predictive maintenance of vehicles and infrastructure. In manufacturing, machine learning optimizes production processes, predicts equipment failures, and improves quality control. The widespread adoption of machine learning across various industries indeed marks a significant paradigm shift, empowering organizations to extract valuable insights from vast amounts of data and make informed decisions.

The success of machine learning relies on the availability of large, high-quality datasets, powerful computing resources, and expertise in algorithm development and data analysis. As the field of machine learning continues to advance, researchers and practitioners are exploring new algorithms, techniques, and applications to tackle increasingly complex and diverse problems. Machine learning holds the potential to revolutionize industries, improve decision-making, and drive innovation in countless domains, shaping the future of technology and society.



**Fig 1.2 Classification of Machine Learning**

The above Fig 1.2 shows the classification of Machine learning. In this machine learning gets classified into Supervised, Unsupervised and Reinforcement Learning techniques. Again, these are classified further like Supervised Learning as classification and regression, Unsupervised Learning as Clustering and the Reinforcement Learning as decision making. The versatility and effectiveness of machine learning make it a transformative force across industries, driving innovation, efficiency, and competitiveness.

Machine learning can be broadly categorized into several types, each with its own characteristics and applications. Here are some of the main types of machine learning are

**Supervised Learning:** In supervised learning, the algorithm learns from labeled data, where each input is associated with a corresponding output label. The algorithm learns to map inputs to outputs, making predictions or classifications based on new, unseen data. Supervised learning tasks include classification, where the algorithm predicts a discrete label or category for each input, and regression, where the algorithm predicts a continuous value.

**Unsupervised Learning:** Unsupervised learning involves learning from unlabeled data, where the algorithm identifies patterns and structures in the data without explicit guidance. Unsupervised learning tasks include clustering, where the algorithm groups similar data points together, and dimensionality reduction, where the algorithm reduces the number of features in the data while preserving important information.

**Semi-Supervised Learning:** Semi-supervised learning combines elements of both supervised and unsupervised learning. It uses a small amount of labeled data and a larger amount of unlabeled data to improve the performance of the learning algorithm. Semi-supervised learning is useful when labeled data is scarce or expensive to obtain.

**Reinforcement Learning:** Reinforcement learning involves training an agent to interact with an environment and learn from feedback in the form of rewards or penalties. The agent takes actions in the environment and learns to maximize its cumulative reward over time. Reinforcement learning is used in applications such as robotics, gaming, and autonomous systems. Reinforcement learning operates on the principle of trial and error, where the agent explores different actions to understand their consequences and adapt its behavior accordingly. Through repeated interactions with the environment, the agent refines its decision-making process, aiming to achieve long-term goals while navigating complex and uncertain environments. This iterative learning approach enables reinforcement learning to tackle tasks that are difficult to solve using traditional algorithmic methods.

**Deep Learning:** Deep learning is a subset of machine learning that uses artificial neural networks with multiple layers (deep architectures) to learn from data. Deep learning algorithms, also known as deep neural networks, can automatically learn hierarchical representations of data, leading to state-of-the-art performance in tasks such as image recognition, natural language processing, and speech recognition.

**Transfer Learning:** Transfer learning involves transferring knowledge from one task or domain to another. Instead of training a model from scratch, transfer learning leverages pre-trained models or learned features to improve performance on a new, related task or domain. Transfer learning is particularly useful when labeled data is limited or when tasks have similar underlying structures.

In this project we are using KNN Algorithm, the details of the knn algorithm are The k-Nearest Neighbors (k-NN) algorithm is a simple and versatile machine learning algorithm used for classification and regression tasks. It is a type of instance-based learning, where the algorithm makes predictions based on the similarity of new data points to existing labeled data points in the training dataset. In the k-NN algorithm, the "k" refers to the number of nearest neighbors considered when making predictions. When a new data point is presented to the algorithm, it calculates the distance between the new point and all points in the training dataset. The k nearest neighbors are then identified based on their distance to the new point. For classification tasks, the most common class among the k nearest neighbors is assigned to the new data point. This is often determined by a majority voting scheme, where each neighbor "votes" for its respective class label, and the class with the highest number of votes is assigned to the new data point. k-NN can be computationally expensive, especially with large datasets, as it requires calculating distances between the new data point and all existing data points in the training set. Techniques such as KD-trees or ball trees are often employed to optimize this process and improve efficiency. While k-NN offers a straightforward approach to classification and regression tasks, careful consideration of parameters and potential computational challenges is essential for achieving optimal results in practical applications.

For regression tasks, the algorithm calculates the average (or weighted average) of the target values of the  $k$  nearest neighbors and assigns this average value as the prediction for the new data point. The choice of the value of  $k$  is a crucial hyperparameter in the  $k$ -NN algorithm. A smaller value of  $k$  may lead to a more flexible model with higher variance but lower bias, while a larger value of  $k$  may result in a smoother decision boundary with lower variance but higher bias.

One of the main advantages of the  $k$ -NN algorithm is its simplicity and ease of implementation. It does not require training a model on the entire dataset, making it particularly useful for online learning scenarios or when dealing with large datasets. Additionally,  $k$ -NN can capture complex decision boundaries and handle nonlinear relationships between features and target variables. However, the  $k$ -NN algorithm can be computationally expensive, especially for large datasets, as it requires calculating distances between the new data point and all points in the training dataset. It is also sensitive to the choice of distance metric and the value of  $k$ , which may need to be optimized through cross-validation.

While the K-Nearest Neighbors (KNN) algorithm is a simple and intuitive classification method, it comes with several disadvantages compared to the Gaussian Naive Bayes (GNB) algorithm. One notable drawback of KNN is its computational inefficiency, particularly when dealing with large datasets. Since KNN requires calculating distances between the query instance and all training instances, the algorithm can become computationally expensive as the dataset size increases. In contrast, GNB is computationally efficient and can handle large datasets more effectively due to its probabilistic nature and assumption of feature independence. KNN is sensitive to irrelevant and noisy features in the dataset. Because KNN relies on a distance metric to determine class labels, noisy or irrelevant features can significantly impact the algorithm's performance. In contrast, GNB assumes feature independence, which can make it more robust to noisy data by effectively ignoring irrelevant features during classification. KNN lacks a natural way to handle categorical features. While it can work with categorical data by encoding them into numerical values, this approach may not always capture the true underlying relationships between categories. GNB naturally handles categorical features by modeling the probability distribution.

## **CHAPTER-2**

### **LITERATURE SURVEY**

Wang et al. (2022) addressed the pressing need for enhanced energy management in smart homes by integrating IoT and machine learning technologies. Their work aligns with a broader trend in smart home research aimed at optimizing energy usage and improving overall efficiency. A significant body of literature exists on the integration of IoT and machine learning in smart home energy management systems. Researchers have explored various aspects of this integration, including real-time energy monitoring, predictive analytics, and adaptive control strategies.

Wilson et al. (2022) investigated optimization techniques for indoor climate control systems in smart homes, which resonate with Wang et al.'s focus on energy management. By leveraging IoT sensors and machine learning algorithms, researchers aim to regulate temperature and humidity levels efficiently while maximizing occupant comfort. Wang et al.'s study adds to a growing body of research that underscores the transformative potential of integrating IoT and machine learning in smart home energy management.

Mia Thompson et al. (2023) conducted a study on user-centric smart home design, emphasizing the importance of human-centered approaches in the development of IoT-enabled systems. By integrating user feedback and preferences into the design process, researchers aim to create more intuitive and personalized smart home solutions that meet the diverse needs of occupants. Additionally, advancements in sensor technology have enabled the development of novel applications in smart home monitoring. For example, Zoe Rodriguez et al. (2021) investigated the use of wearable sensors for health monitoring in smart homes, leveraging machine learning techniques to analyze physiological data and provide personalized healthcare insights to users.

The integration of edge computing with IoT and machine learning has emerged as a promising approach to enhance the scalability and responsiveness of smart home systems. Research by Gabriel Perez et al. (2022) explores edge intelligence frameworks for real-time data processing and decision-making in smart homes,

enabling faster response times and reducing reliance on cloud-based services. Surrounding smart home monitoring systems using IoT and machine learning is vast and diverse, encompassing a wide range of topics from energy management and security to user-centric design and edge computing. This body of research reflects the growing interest and investment in creating intelligent and interconnected living environments that enhance comfort, convenience, and sustainability for residents.

Mario Collotta et al (2022) has proposed system present an Artificial Neural Network (ANN) as maintain for a Home Energy Management (HEM) arrangement base on Bluetooth low energy, called BluHEMS. The objective of infrastructure technology is to realize an extensive energy savings, in order to cut greenhouse gas emissions and to reach effectual ecological security in more than a few contexts, counting infrastructure, developed, transport, buildings, electricity generation and delivery.

Siyun Chen et al (2021) gives a human-centric Smart Home Energy Management System (SHEM) with the intention of mechanism at the “butler” plane. In this proposed system a human-centric smart home construction by the combination of considerate the behaviors of person being, infer the user’s weight and utilization preference and optimally running the energy plans in smart home. The SHE system can be comprehensive for respond to a variety of require response signal and contribution expensive maintain for programming and result make at all levels of usefulness based on the people-centric framework.

Hassan et al. (2022) provided an overview of building energy consumption in Malaysia, offering valuable context for understanding regional energy usage patterns and challenges. Their research highlights the importance of addressing energy efficiency and sustainability concerns in residential and commercial buildings.

Kim et al. (2021) introduced an IoT-based home energy management system designed to operate over dynamic home area networks. This research addresses the need for efficient energy management in residential settings by leveraging IoT technologies to monitor and control energy consumption. This is particularly crucial given the increasingly interconnected and heterogeneous nature of modern households, where energy usage can fluctuate based on factors such as occupancy, weather conditions.

Aburukba et al. (2022) further explored IoT-based energy management for residential areas, presenting their findings at the IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW). Their work contributes to the growing body of research on IoT applications in energy management, emphasizing the potential for smart solutions to optimize energy usage in homes.

Hou et al. (2022) discussed the architecture and implementation of IoT cloud systems, shedding light on the technical aspects of IoT infrastructure for data storage, processing, and analysis. This study provides valuable insights into the design and deployment of IoT solutions in various domains, including home energy management.

Tozlu et al. (2019) explored the practical applications of Wi-Fi-enabled sensors for IoT deployments, focusing on their role in collecting and transmitting data for various IoT applications. Their study contributes to the understanding of sensor technologies and their integration into IoT ecosystems.

Han et al. (2021) presented a smart home energy management system that integrates renewable energy sources using ZigBee and PLC technologies. This research demonstrates the potential for integrating diverse energy sources and communication protocols to create comprehensive energy management solutions for smart homes.

Recent research in smart home energy management integrates IoT and machine learning, as exemplified by studies from Wang et al. (2022) and Wilson et al. (2022). Mia Thompson et al. (2023) emphasize user-centric design, while Gabriel Perez et al. (2022) explore edge computing's role. Mario Collotta et al. (2022) propose BluHEMS for energy savings, and Siyun Chen et al. (2021) introduce SHEM for human-centric energy management. Regional perspectives, such as Hassan et al. (2022)'s work on Malaysian energy consumption, and technical insights like Hou et al. (2022)'s IoT cloud systems architecture, further enrich the field. Collectively, these studies drive forward the development of sustainable, efficient, and user-friendly smart home solutions. By integrating insights from various disciplines and addressing regional-specific challenges, researchers continue to drive progress towards creating smarter, more energy-efficient, and environmentally conscious homes for the future.

## **CHAPTER-3**

### **AIM AND SCOPE OF THE PRESENT INVESTIGATION**

#### **3.1 AIM**

The aim of a smart home monitoring system leveraging Internet of Things (IoT) and machine learning technologies is to create an intelligent and responsive living environment that enhances convenience, efficiency, and security for occupants. By integrating IoT devices such as sensors, actuators, and smart appliances with sophisticated machine learning algorithms, these systems seek to achieve real-time monitoring, analysis, and optimization of various aspects of the home environment.

The overarching goal is to enable proactive management of energy consumption, climate control, and security, while also catering to individual preferences and lifestyle patterns. Through continuous data collection, analysis, and learning, smart home monitoring systems can adapt dynamically to changing conditions, anticipate user needs, and autonomously optimize environmental settings.

Moreover, by leveraging machine learning for anomaly detection and predictive analytics, these systems can enhance security measures, detect potential threats, and mitigate risks in real-time. Ultimately, the aim is to empower homeowners with greater control, comfort, and peace of mind, while simultaneously reducing energy waste, enhancing sustainability, and improving overall quality of life.

The convergence of IoT and machine learning technologies in smart home monitoring systems represents a transformative approach to residential living, offering unprecedented levels of convenience, efficiency, security, and sustainability. Through continuous innovation and interdisciplinary collaboration, researchers and practitioners are driving the evolution of smart homes into intelligent, adaptive, and harmonious living environments. As research and innovation in this field continue to advance, the vision of intelligent, adaptive, and harmonious living environments becomes increasingly attainable, transforming the way we experience and interact with our homes. By empowering homeowners to make informed decisions and adopt eco-friendly practices, these systems contribute to reducing carbon footprints.

### **3.2 SCOPE OF THE PROJECT**

A smart home monitoring system leveraging Internet of Things (IoT) and machine learning technologies constitutes a multifaceted solution designed to revolutionize residential living. Its scope encompasses a wide array of functionalities and applications aimed at optimizing various aspects of the home environment, thereby enhancing convenience, efficiency, security, and sustainability for occupants. At the forefront of its capabilities lies energy management, whereby the system meticulously monitors and controls energy usage across different devices and appliances. By analyzing consumption patterns and implementing optimization strategies, it aims to reduce waste, lower utility costs, and promote eco-friendly practices within the household.

Moreover, climate control stands as a pivotal component, whereby the system regulates heating, ventilation, and air conditioning (HVAC) systems based on occupancy, weather conditions, and user preferences. This ensures optimal indoor comfort levels while simultaneously minimizing energy expenditure and environmental impact. Complementing this function is the system's robust security and surveillance features, which deploy advanced sensors and machine learning algorithms to detect and deter intrusions, unauthorized access, and suspicious activities. Through intelligent analysis and real-time alerts, it provides homeowners with peace of mind and a heightened sense of security.

The concept of home automation further enriches the system's capabilities by automating routine tasks and actions such as lighting control, appliance scheduling, and irrigation systems. This not only enhances convenience for occupants but also contributes to overall energy efficiency and resource conservation. Additionally, predictive maintenance emerges as a proactive approach to managing the health and performance of home appliances and systems. By leveraging machine learning algorithms to detect potential faults or malfunctions before they occur, the system enables timely interventions, minimizing downtime and maintenance costs. Home automation facilitates seamless integration with external systems and services, such as weather forecasts or energy tariffs, to further optimize operations and energy usage. For example, smart irrigation systems can adjust watering schedules based on weather predictions, conserving water resources while maintaining healthy gardens and landscapes.

Personalization lies at the heart of the user experience, with machine learning algorithms continuously learning user preferences, behavior patterns, and daily routines. This adaptive approach allows the system to tailor environmental settings and automation rules to cater to individual needs, thereby enhancing comfort and satisfaction for occupants. Meanwhile, the system's data analytics capabilities empower homeowners with actionable insights derived from the vast amount of sensor data collected from IoT devices. By identifying trends and optimizing system performance over time, it facilitates informed decision-making and continual improvement.

Interoperability and integration remain critical considerations, ensuring compatibility and seamless interaction with existing smart home devices, platforms, and ecosystems. This fosters flexibility, scalability, and customization, enabling homeowners to expand and enhance their smart home environment as needed. Furthermore, accessibility and user interface design play pivotal roles in ensuring effortless interaction and control of the system. Intuitive interfaces such as mobile apps, voice assistants, and web dashboards empower users to effortlessly manage their smart home environment with ease. In the ever-evolving landscape of technology, continuous innovation and evolution are imperative for smart home monitoring systems. By staying abreast of emerging technologies, standards, and best practices in IoT and machine learning, these systems can adapt and evolve to meet evolving user needs and industry trends. Through ongoing research, development, and collaboration, they promise to redefine the concept of modern living, ushering in an era of intelligent, responsive, and harmonious homes that enrich the lives of occupants while promoting sustainability and efficiency.

It is essential to highlight their role in promoting sustainability and environmental responsibility. Beyond energy management, these systems actively contribute to reducing carbon footprints by optimizing resource utilization and minimizing waste. Through real-time monitoring and analysis of energy consumption, water usage, and other resources, they empower homeowners to make informed decisions that align with eco-conscious practices. Additionally, by integrating renewable energy sources such as solar panels and wind turbines, smart home monitoring systems further reduce reliance on traditional energy grids, fostering greater energy independence and resilience.

## **CHAPTER-4**

### **EXPERIMENTAL METHODS**

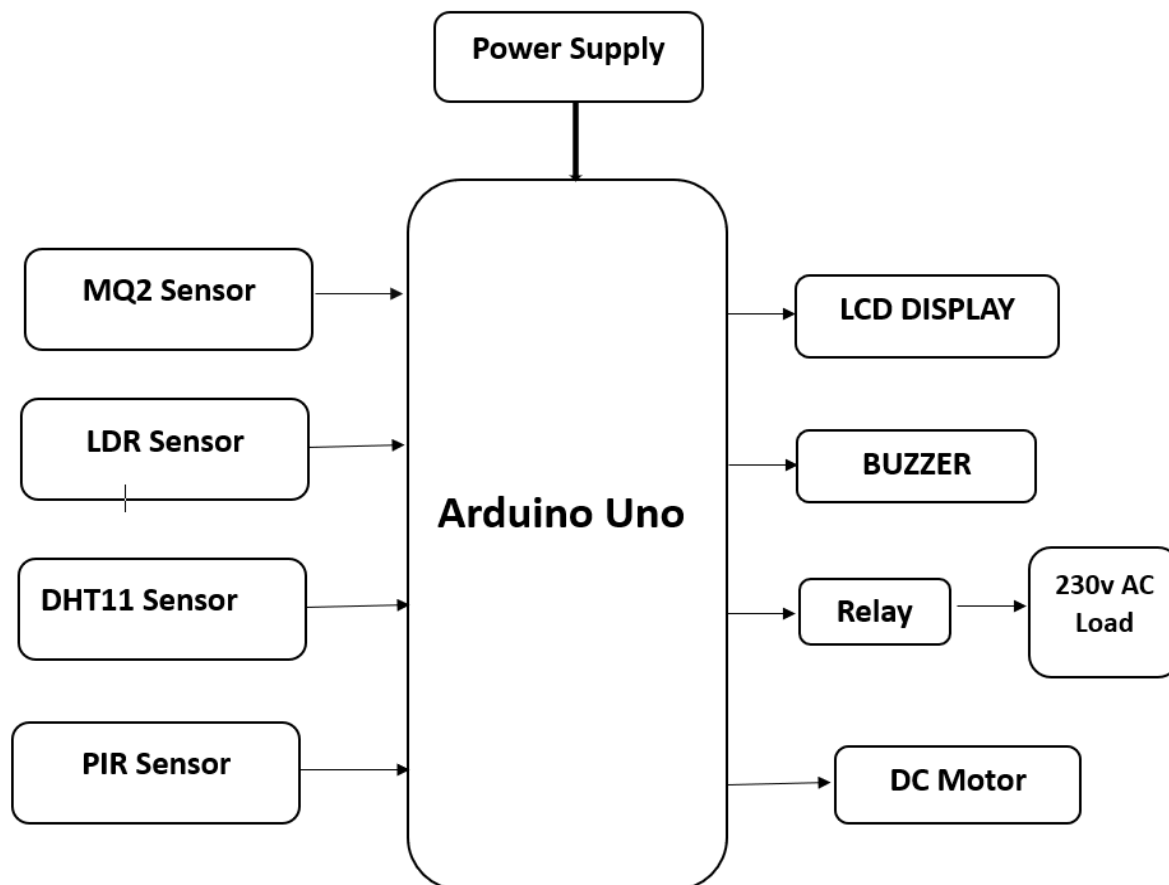
#### **4.1 EXISTING SYSTEM**

In existing systems we don't have any automatic monitoring controlling of building. In existing system, manual monitoring of building are required which is dangerous and it requires lots of time to monitor and controlling Building Energy Management systems are critical in many regions, being used over several decades. In the Existing System, meter reading taken manually Errors occurred due to less concentration Meter reading not accurate Huge Manpower needed Delayed work due to external conditions Power cut manually due to lack of payment. In proposed system Meter reading taken automatically using GSM technology Power cut achieved from EB office through wireless work happen at any time.

#### **4.2 PROPOSED SYSTEM**

In proposed system Machine learning method has been applied to analyze the data. Word embedding method in unsupervised learning using large dataset. The feature set is integrated into a Linear Regression for training and predicting classification labels. Linear Regression architecture is capable of overcoming the performance of traditional machine learning models. For easy access the server maintains a database for each node, and hence each node will have a unique id for addressing it. Proposed system, the various building data are collected from sensors like co2, LDR, DHT11, motion sensors. The data collected by the IOT sensors are given to the Arduino Microcontroller. Based on PIR and LDR then lights will ON. Based on DHT11 Values the speed of fan will be changed. Based on CO2 Values a DC Motor will rotate. The values are continuously sent to the cloud server where user can continuously checks the sensors parameters. The sensors are placed in acceptable places of buildings to gather needed features.

### 4.3 PROPOSED BLOCK DIAGRAM



**Fig 4.1 Proposed Block Diagram**

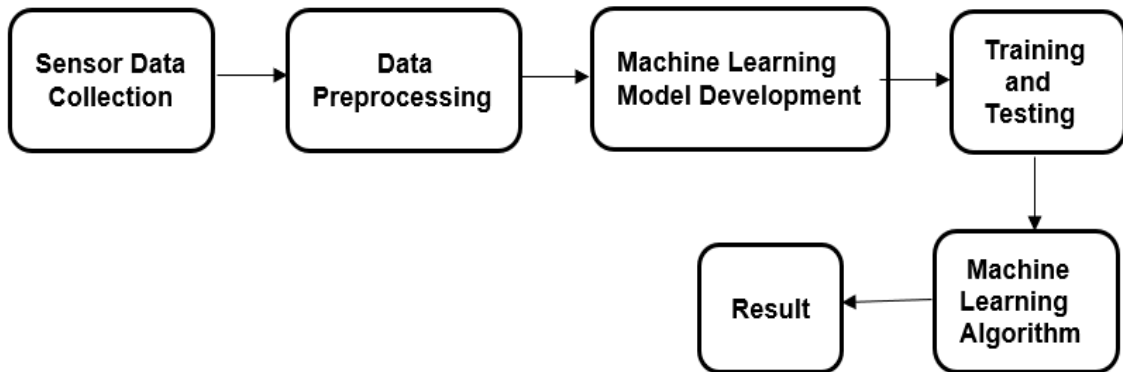
Fig 4.1 shows the proposed block diagram and it provides a comprehensive view of the interconnected components and their functionalities within our project. Our project is designed to offer a sophisticated environment monitoring and control system leveraging the capabilities of an Arduino Uno microcontroller. By integrating a diverse array of sensors and actuators, including LDR (Light Dependent Resistor), PIR (Passive Infrared), DHT11 (Temperature and Humidity), MQ2 (Gas), as well as supplementary elements such as a power supply, LCD display, DC motor, and buzzer, we've created a versatile platform capable of addressing various environmental monitoring and control needs.

In our sensor data acquisition system, a network of sensors continuously monitors environmental parameters, collecting vital data that is transmitted to the Arduino Uno

for processing. This collaborative effort ensures a steady stream of information essential for informed decision-making. Leveraging programmed algorithms and decision logic, the Arduino Uno analyzes this incoming data, assessing it against predefined thresholds and user-defined settings to determine appropriate responses. These responses may involve activating actuators, triggering alarms, or updating the user interface to reflect the system's current status and any relevant information. Upon reaching a decision, the Arduino Uno swiftly issues commands to the relevant actuators, such as a DC motor for physical adjustments or a buzzer for audible alerts. This actuation process is integral to the system's functionality, ensuring that responses are promptly executed to maintain optimal environmental conditions or address emerging issues. Concurrently, the Arduino Uno updates the LCD display, providing users with real-time feedback on the system's actions and any pertinent alerts or notifications. This bidirectional communication channel fosters user engagement and empowerment, allowing users to interact with the system, access menus, adjust settings, or acknowledge alerts according to their preferences and requirements.

Through the seamless integration of these components, our project exemplifies the potential of Arduino-based systems in creating smart and responsive environments. Whether deployed in homes, offices, or industrial settings, the versatility and adaptability of our system enable it to address diverse needs, from energy conservation and comfort optimization to safety and security enhancement. With further refinements and expansions, our platform holds promise for applications in the Internet of Things (IoT), smart buildings, and beyond, contributing to the advancement of intelligent and sustainable technologies.

As technology continues to evolve, the importance of creating systems that can adapt to changing environments and user needs becomes increasingly evident. Our project represents a step forward in this direction, showcasing the capabilities of Arduino-based systems to efficiently monitor, analyze, and respond to environmental data in real time and represents a significant advancement in creating adaptive and responsive systems that can effectively meet the evolving needs and challenges of modern environments. By harnessing the power of Arduino-based technologies, we pave the way for the development of innovative solutions that enhance efficiency, sustainability, and user experience across diverse domains.



***Fig 4.2 Block Diagram of Machine Learning***

The Fig 4.2 provides a holistic view of our project, integrating both traditional sensor-driven control mechanisms and machine learning algorithms for enhanced decision-making capabilities. Leveraging an Arduino Uno microcontroller as the core processing unit, our system encompasses sensor data collection, data processing, machine learning model development, training and testing phases, machine learning algorithms (including Gaussian Naive Bayes and K-Nearest Neighbors), and result analysis.

**Sensor Data Collection:** The sensors continuously gather environmental data, including light intensity, motion, temperature, humidity, and gas levels. This raw data serves as the input for subsequent processing and analysis stages.

**Data Processing:** The collected sensor data undergoes preprocessing to remove noise, handle missing values, and normalize features. Preprocessed data is then fed into the machine learning pipeline for model training and testing.

**Machine Learning Model Development:** Machine learning models are developed based on the processed sensor data and project objectives. Feature selection, model architecture design, and hyperparameter tuning are conducted to optimize model performance.

Training and Testing: The machine learning models are trained using labeled data to learn patterns and relationships within the dataset. Subsequently, the trained models are tested on unseen data to assess their generalization capabilities and performance metrics.

Machine Learning Algorithms: Various machine learning algorithms are applied to the dataset, each with its unique approach to learning and making predictions. The selection of algorithms depends on the nature of the task, data characteristics, and performance requirements.

Machine Learning Algorithms (Gaussian Naive Bayes, KNN): Gaussian Naive Bayes and K-Nearest Neighbors are two of the machine learning algorithms employed in our system. Gaussian Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem with the assumption of independence among predictors. KNN is a non-parametric classification algorithm that classifies data points based on the majority class of their  $k$  nearest neighbors.

The performance of trained machine learning models is evaluated using standard evaluation metrics. This analysis provides insights into the effectiveness of the models and informs decision-making regarding model deployment or refinement. By integrating machine learning capabilities into our project architecture, we augment traditional sensor-driven control mechanisms with predictive and adaptive functionalities. This hybrid approach enables the system to not only react to environmental stimuli but also anticipate future conditions and adapt proactively. Through continuous learning and refinement, our system strives to optimize performance, enhance user experience, and contribute to the advancement of intelligent automation technologies. In our project architecture, the incorporation of machine learning algorithms, specifically Gaussian Naive Bayes and K-Nearest Neighbors (KNN), brings a new dimension to our environmental monitoring and control system. The utilization of these algorithms enables our system to go beyond simple sensor data collection and traditional rule-based decision-making.

Gaussian Naive Bayes, a probabilistic classification algorithm, leverages Bayes' theorem to classify data based on the likelihood of occurrence of certain events. Despite its "naive" assumption of independence among predictors, Gaussian Naive Bayes provides a robust framework for classifying environmental conditions such as air quality or occupancy status. On the other hand, K-Nearest Neighbors algorithm operates on the principle of similarity, classifying data points based on the majority class of their nearest neighbors in the feature space. By employing KNN, our system can classify environmental data points based on their proximity to previously labeled instances, allowing for more nuanced decision-making in real-time. Integrating these machine learning algorithms into our architecture not only enhances the system's ability to classify and predict environmental conditions but also enables it to adapt dynamically to evolving contexts. Through continuous learning and refinement, our system strives to optimize performance, improve user experience, and contribute to the advancement of intelligent automation technologies in various domains.

#### **4.4 HARDWARE REQUIREMENT**

##### **4.4.1 *Arduino uno***

Arduino is a tool for making computers that can sense and control more of the physical world than your desktop computer. It's an open-source physical computing platform based on a simple microcontroller board, and a development environment for writing software for the board.

Arduino can be used to develop interactive objects, taking inputs from a variety of switches or sensors, and controlling a variety of lights, motors, and other physical outputs. Arduino projects can be stand-alone, or they can be communicate with software running on your computer. The boards can be assembled by hand or purchased preassembled; the open-source IDE can be downloaded for free.

The Arduino programming language is an implementation of Wiring, a similar physical computing platform, which is based on the Processing multimedia programming environment. This integration of Arduino programming language with Wiring, and its roots in the Processing multimedia programming environment, lays a robust foundation for building interactive and responsive systems. Wiring's simplicity and ease of use make it accessible to beginners, while its flexibility and extensibility cater to the needs of advanced users and professionals alike.

The Arduino microcontroller is an easy to use yet powerful single board computer that has gained considerable traction in the hobby and professional market. The Arduino is open-source, which means hardware is reasonably priced and development software is free.

This guide covers the Arduino Uno board (Spark fun DEV-09950, \$29.95), a good choice for students and educators. With the Arduino board, you can write programs and create interface circuits to read switches and other sensors, and to control motors and lights with very little effort.



**Fig 4.3 Arduino Board**

The above Fig 4.3 shows the Arduino UNO Board which is used for control applications and it is a programmable device. The Arduino programming language is a simplified version of C/C++. If you know C, programming the Arduino will be familiar. If you do not know C, no need to worry as only a few commands are needed to perform useful functions. The Arduino Uno is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The onboard USB connection facilitates easy programming and communication with a computer, while the availability of power options, including USB, AC-to-DC adapters, and batteries, ensures flexibility in deployment scenarios.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

Microcontroller ATmega328

Operating Voltage 5V

Input Voltage (recommended) 7-12V

Input Voltage (limits) 6-20V

Digital I/O Pins 14 (of which 6 provide PWM output)

Analog Input Pins 6

DC Current per I/O Pin 40 mA

DC Current for 3.3V Pin 50 mA

Flash Memory 32 KB (ATmega328) of which 0.5 KB used by bootloader

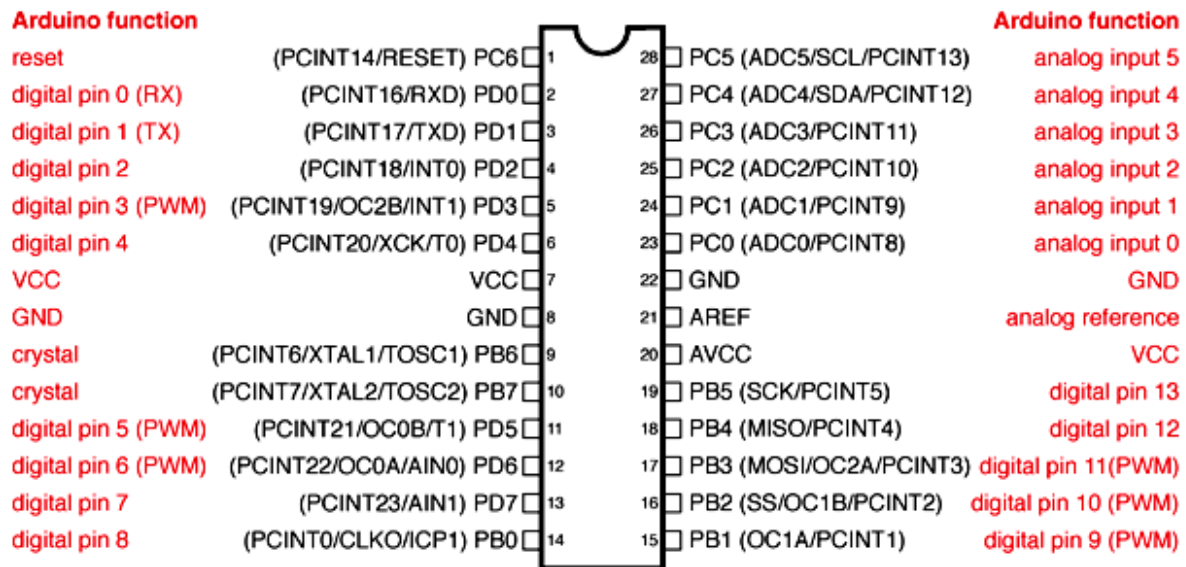
SRAM 2 KB (ATmega328)

EEPROM 1 KB (ATmega328)

Clock Speed 16 MHz

#### Pin Configuration

The Arduino Uno is a popular microcontroller board with a set of digital and analog input/output pins that can be used for various purposes, including interfacing with sensors, displays, and other electronic components. The Arduino Uno features 14 digital I/O pins, each capable of being configured as either inputs or outputs. Additionally, six of these pins support Pulse Width Modulation output, allowing for precise control over analog-like outputs such as LED brightness or motor speed. The board also includes six analog input pins, designated for reading analog voltage values from sensors or other analog devices. Specific pins like RX and TX facilitate serial communication, enabling the Arduino Uno to communicate with other devices via UART. The board also includes six analog input pins, designated for reading analog voltage values from sensors or other analog devices. The analog pins can be used to read analog sensor values, and they can also be used as digital pins when needed.



The above Fig 4.4 shows the Arduino UNO pin configuration and there are 28 pins in Arduino UNO.

Digital Pins 0 to 13: These pins can be used for both digital input and output. Pins 0 and 1 are also used for serial communication (RX and TX).

Analog Pins A0 to A5: These pins are used for analog input. They can also be used as digital input/output pins.

5V: Provides a 5-volt power supply for external components.

GND (Ground): Ground pins for completing electrical circuits.

## Communication Pins

RX (Receive): Used for serial communication and receiving data.

TX (Transmit): Used for serial communication and transmitting data.

SDA (Analog Pin A4): Used for I2C communication (TWI - Two-Wire Interface).

SCL (Analog Pin A5): Also used for I2C communication.

The Arduino Uno is a popular microcontroller board with a set of digital and analog input/output pins that can be used for various purposes, including interfacing with sensors, displays, and other electronic components.

## Other Pins

Reset: Used to reset the microcontroller.

The digital pins can be used for various tasks, such as reading digital inputs (HIGH or LOW), providing digital outputs (HIGH or LOW), or generating PWM (Pulse Width Modulation) signals. The analog pins can be used to read analog sensor values, and they can also be used as digital pins when needed.

### **4.4.2 MQ2 Sensor**

The MQ-2 sensor is a type of gas sensor commonly used for detecting and measuring various types of combustible gases, as well as smoke and other volatile organic compounds (VOCs) in the air. Developed by the Hanwei Electronics Group Corporation, the MQ-2 sensor is widely utilized in a range of applications including gas leak detection, fire prevention systems, air quality monitoring, and industrial safety equipment.

At its core, the MQ-2 sensor operates based on the principle of gas detection through the variation of electrical conductivity in the presence of target gases. The sensor consists of a sensitive semiconductor element composed of a tin dioxide ( $\text{SnO}_2$ ) thin film, which acts as a gas-sensing material. When exposed to the target gas molecules, such as methane, propane, alcohol, or carbon monoxide, the conductivity of the  $\text{SnO}_2$  film changes proportionally to the gas concentration in the air.

The MQ-2 sensor is typically integrated into electronic circuits along with a heating element and a load resistor. The heating element, often a small coil of Nichrome wire, is used to raise the operating temperature of the  $\text{SnO}_2$  thin film to a level where it becomes sensitive to gas molecules. As the gas concentration in the surrounding environment fluctuates, the resistance of the  $\text{SnO}_2$  film changes accordingly, leading to variations in the output voltage across the load resistor. This output voltage can be measured and analyzed to determine the concentration of the target gas.

One of the key advantages of the MQ-2 sensor is its low cost, compact size, and ease of integration into electronic systems. Moreover, it offers rapid response times and high sensitivity to a wide range of combustible gases, making it suitable for diverse applications. However, it's important to note that the MQ-2 sensor has limitations, including cross-sensitivity to certain gases, susceptibility to environmental factors.



**Fig 4.5 MQ2 Sensor**

The above Fig 4.5 shows the MQ2 sensor. The MQ-2 sensor remains a popular choice for gas detection and monitoring applications due to its affordability, versatility, and reliability. When properly calibrated and integrated into a monitoring system, the MQ-2 sensor can contribute significantly to enhancing safety, preventing hazards, and ensuring air quality in various indoor and outdoor environments.

#### **4.4.3 LDR Sensor**

A Light Dependent Resistor (LDR), also known as a photoresistor, is a passive electronic component that exhibits a change in resistance based on the intensity of light incident upon it. LDRs are widely used in various applications where detection or measurement of light levels is required, such as in automatic lighting control systems, photographic light meters, and solar panels.

The fundamental principle behind the operation of an LDR is the variation of its resistance with changing light levels. Typically, an LDR is composed of a semiconductor material whose conductivity increases as the intensity of incident light increases. When exposed to darkness or low light conditions, the resistance of the LDR is high, limiting the flow of electric current. Conversely, when exposed to bright light, the resistance decreases, allowing more current to pass through.

The construction of an LDR usually involves a semiconductor material such as cadmium sulfide (CdS) or cadmium selenide (CdSe) sandwiched between two electrodes. The semiconductor material's conductivity is influenced by the amount of light absorbed by its atoms, which in turn affects the number of charge carriers (electrons and holes) available for conduction. The construction and behavior of LDRs are based on the principles of semiconductor physics, where light absorption influences the conductivity of the material, thereby modulating its resistance.

In practical applications, an LDR is typically connected to a circuit in a voltage divider configuration, where it acts as one of the resistors. As the light level changes, the resistance of the LDR changes accordingly, causing a corresponding change in the voltage across it. This voltage variation can be measured and utilized by the connected circuitry for various purposes, such as triggering a relay to control lighting, adjusting camera exposure settings, or regulating the operation of solar-powered devices.

One of the key advantages of LDRs is their simplicity, low cost, and ease of integration into electronic circuits. They require no external power source and exhibit a linear response to changes in light intensity over a wide range of wavelengths. However, it's important to note that LDRs have limitations, including relatively slow response times and susceptibility to temperature variations.



***Fig 4.6 LDR Sensor***

The above Fig 4.6 shows the LDR sensor. LDRs are valuable components in a wide range of light sensing and control applications, offering a cost-effective solution for monitoring and adjusting light levels in diverse environments. Their versatility and reliability make them indispensable in numerous consumer, industrial, and scientific applications where light detection and measurement are essential. Its functionality relies on its ability to alter resistance based on the intensity of light falling upon it. By integrating an LDR with an Arduino microcontroller, users can create sophisticated light-sensing systems capable of adapting to varying environmental conditions. Through simple wiring and programming, the Arduino reads analog voltage values from the LDR, translating them into actionable data for further processing. Whether used to automate lighting systems, regulate environmental controls, or trigger responsive behaviors, the versatility of the LDR sensor empowers creators to design innovative solutions for diverse applications.

#### **4.4.4 PIR Sensor**

A Passive Infrared (PIR) sensor is a type of motion sensor that detects changes in infrared radiation emitted by objects within its field of view. PIR sensors are commonly used in various applications such as security systems, automatic lighting control, and occupancy detection.

The principle of operation of a PIR sensor is based on the detection of heat radiated by objects in its surroundings. Each object with a temperature above absolute zero emits infrared radiation, and this radiation is detected by the sensor. The PIR sensor consists of a pyroelectric material that generates an electrical signal when exposed to infrared radiation. When an object moves within the sensor's field of view, it causes a change in the pattern of infrared radiation reaching the sensor, leading to a detectable signal change.

The key components of a PIR sensor include the pyroelectric sensor element, a lens to focus the infrared radiation onto the sensor, and electronic circuitry for signal processing and amplification. The sensor element typically consists of a thin film of pyroelectric material, such as lithium tantalate or polyvinyl fluoride, which generates an electric charge in response to changes in temperature caused by infrared radiation.

In practical applications, PIR sensors are often used in combination with other devices, such as microcontrollers or relays, to trigger specific actions based on detected motion. For example, in security systems, a PIR sensor can be connected to an alarm system to activate an alarm when motion is detected in a protected area. In automatic lighting control systems, PIR sensors can be used to turn on lights when motion is detected and turn them off after a certain period of inactivity, thereby conserving energy.

One of the key advantages of PIR sensors is their low cost, simplicity, and effectiveness in detecting motion without direct contact with the object. However, it's important to note that PIR sensors have limitations, such as their inability to detect stationary objects or objects that do not emit infrared radiation, such as glass or plastic. PIR sensors are highly effective in detecting motion without the need for direct contact with the object being monitored. This non-contact detection method not only simplifies installation but also eliminates the risk of wear and tear associated with physical contact sensors.



***Fig 4.7 PIR Sensor***

The above Fig 4.7 shows the PIR sensor. PIR sensors are valuable components in motion detection and occupancy sensing applications, offering a reliable and cost-effective solution for enhancing security, convenience, and energy efficiency in various environments.

#### **4.4.5 DHT11 Sensor**

The DHT11 is a low-cost digital temperature and humidity sensor module commonly used in various applications where monitoring environmental conditions is necessary. Developed by the Chinese company Aosong Electronics, the DHT11 sensor provides accurate readings of temperature and relative humidity with relatively simple interfacing requirements.

The DHT11 sensor module consists of a capacitive humidity sensor, a thermistor for temperature measurement, and a small microcontroller that digitizes the sensor outputs and communicates with external devices via a single-wire digital interface. The sensor utilizes a proprietary digital signal transmission protocol to send temperature and humidity data to a microcontroller or microprocessor.

Operating the DHT11 sensor is straightforward: it requires only a single digital input/output pin for communication with the host device. The microcontroller within the sensor module converts the analog signals from the temperature and humidity sensors into digital data and transmits it serially using a simple, time-based protocol. The sensor communicates with the host device using a simple, single-wire digital interface, which requires only one digital input/output pin on the microcontroller.

One of the key features of the DHT11 sensor is its affordability and ease of use, making it accessible for hobbyists, students, and makers. Additionally, it offers relatively good accuracy and stability for many general-purpose applications. However, it's essential to note that the DHT11 sensor has limitations, including a relatively slow response time and lower accuracy compared to more advanced and expensive sensors.

The DHT11 sensor is commonly utilized in a wide range of projects and products, including weather stations, environmental monitoring systems, HVAC (Heating, Ventilation, and Air Conditioning) control systems, and agricultural applications. Its compact size, low power consumption, and simplicity make it suitable for integration into various electronic devices and IoT (Internet of Things) projects where temperature and humidity monitoring are essential.



***Fig 4.8 DHT11 Sensor***

Fig 4.8 shows that the DHT11 sensor. It provides a cost-effective solution for measuring temperature and humidity in applications where high precision is not critical, making it a popular choice for DIY enthusiasts and educational purposes. The integration of the DHT11 sensor into our system enhances its capabilities, allowing for precise monitoring of environmental conditions crucial for maintaining desired settings. Whether deployed in residential, commercial, or industrial settings, the DHT11 sensor provides invaluable insights into temperature and humidity levels, enabling the system to respond effectively to changing conditions in real time. User interaction with the system is facilitated through the LCD display, where users can access temperature and humidity readings, adjust settings, and respond to alerts or notifications as needed. User interaction with the system is facilitated through the LCD display, providing convenient access to temperature and humidity readings, as well as options to adjust settings and respond to alerts or notifications.

#### **4.4.6 LCD Display**

LCD (Liquid Crystal Display) with a 16x16 character matrix. While such displays are not as common as standard 16x2 or 20x4 character LCDs, they are still available and can be used in various applications where a larger display area or higher resolution is required.

A 16x16 character LCD display would typically consist of 16 rows and 16 columns of pixels, allowing for the display of more complex graphics, custom symbols, or larger characters compared to standard character LCDs. These displays are often used in applications such as graphical user interfaces (GUIs), gaming consoles, digital clocks, and instrumentation panels.

Interfacing a 16x16 character LCD display with a microcontroller or other control circuitry would involve similar principles as with smaller character LCDs, but with the additional complexity of managing a larger number of pixels and addressing individual cells within the display matrix.

To drive a 16x16 character LCD display, you would typically need a suitable controller or driver IC that supports the desired resolution and interface protocol (such as parallel or serial communication). Additionally, you may need to develop or use existing software libraries to facilitate communication with the display and generate the necessary pixel data for displaying text, graphics, or custom symbols.

When designing applications with a 16x16 character LCD display, it's essential to consider factors such as display refresh rate, memory requirements for storing pixel data, and the computational overhead for generating and updating display content in real-time. The LCD display plays a crucial role in providing users with real-time feedback and enabling interaction with the system. Integrated seamlessly with the Arduino Uno and other components, the LCD display serves as a user-friendly interface through which users can access important information, adjust settings, and respond to alerts or notifications. The refresh rate of the LCD display determines how often the display updates its content, which is crucial for providing smooth and responsive feedback to the user. It's essential to choose a display with a fast refresh rate to minimize lag and maintain smooth transitions.



***Fig 4.9 16x2 LCD Display***

Fig 4.9 shows that the standard character LCDs, 16x16 character LCD displays offer increased flexibility and visual appeal for applications requiring larger or more detailed graphical output. With the right hardware and software integration, they can enhance user interaction and display capabilities in various electronic devices and systems.

#### **4.4.7 DC Motor**

A DC (Direct Current) motor is an electrical device that converts electrical energy into mechanical energy. It operates based on the principle of electromagnetic induction, where the interaction between magnetic fields and current-carrying conductors generates rotational motion. The basic construction of a DC motor consists of two main parts: the stator and the rotor. The stator is the stationary part of the motor and typically consists of one or more electromagnets arranged in a circular fashion around the rotor. These electromagnets produce a magnetic field when an electric current is passed through them.

The rotor, on the other hand, is the rotating part of the motor and is positioned within the magnetic field produced by the stator. The rotor usually comprises a coil of wire (known as the armature) mounted on a shaft. When a current is passed through the armature coil, it becomes an electromagnet and interacts with the magnetic field generated by the stator. Field windings connected in series, allowing them to produce high torque at low speeds. They are commonly used in applications requiring high starting torque. This interaction between the magnetic fields causes the rotor to rotate, thus converting electrical energy into rotational mechanical energy.

The interaction between the magnetic fields of the stator and the rotor produces a torque, causing the rotor to rotate. The direction of rotation depends on the polarity of the magnetic fields and the direction of the current flow through the armature coil. By controlling the magnitude and direction of the current supplied to the motor, the speed and direction of rotation can be varied as desired. DC motors are widely used in a variety of applications due to their simplicity, reliability, and controllability. They are commonly found in household appliances, industrial machinery, automotive systems, robotics, and various other devices requiring rotational motion.

There are several types of DC motors, including:

**Brushed DC Motors:** These motors use brushes and a commutator to switch the direction of current flow in the armature coil, allowing for continuous rotation. They are simple and inexpensive but require periodic maintenance due to brush wear.

**Brushless DC Motors (BLDC):** BLDC motors use electronic commutation instead of brushes and a commutator, resulting in smoother operation and reduced maintenance requirements. They are commonly used in applications requiring high efficiency and reliability.

**Permanent Magnet DC Motors:** These motors have a permanent magnet rotor and are often more compact and efficient than other types of DC motors. They are widely used in portable devices and small appliances.

**Series-Wound DC Motors:** Series-wound motors have the armature and field windings connected in series, allowing them to produce high torque at low speeds. They are commonly used in applications requiring high starting torque, such as electric vehicles and industrial machinery.

DC motors find widespread use in various applications, including robotics, industrial automation, electric vehicles, and household appliances. They are valued for their simplicity, reliability, and controllability, making them suitable for a wide range of tasks requiring rotational motion. It operates on the principle of electromagnetism, where the interaction between the magnetic field produced by the motor's internal components. This simplicity makes them particularly attractive for applications.

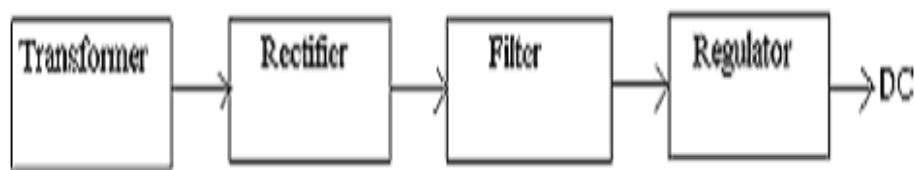


**Fig 4.10 DC Motor**

The above Fig 4.10 shows the DC Motor. DC motors play a crucial role in various industries and applications, providing efficient and reliable means of converting electrical energy into mechanical motion.

#### **4.4.8 Power Supply**

All electronic circuits' works only in low DC voltage, so we need a power supply unit to provide the appropriate voltage supply for their proper functioning. This unit consists of transformer, rectifier, filter & regulator. AC voltage of typically 230volts rms is connected to a transformer voltage down to the level to the desired ac voltage. A diode rectifier that provides the full wave rectified voltage that is initially filtered by a simple capacitor filter to produce a dc voltage. This resulting dc voltage usually has some ripple or ac voltage variation. A regulator circuit can use this dc input to provide dc voltage that not only has much less ripple voltage but also remains the same dc value even the dc voltage varies somewhat, or the load connected to the output dc voltages changes.



**Fig 4.11 General Block of Power Supply Unit**

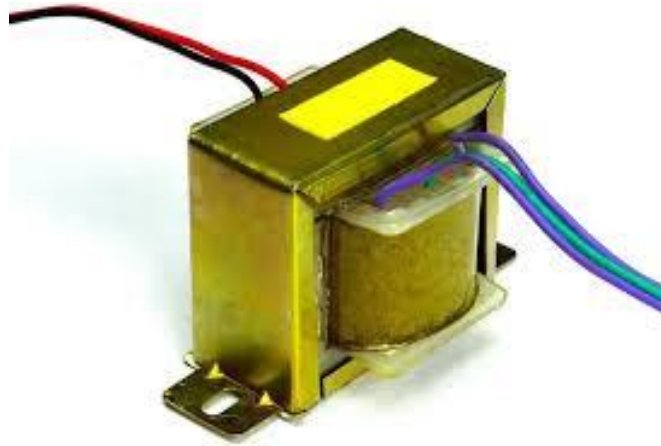
The above Fig 4.11 shows the Block diagram of power supply unit. It contains the transformer which is used to regulate the voltage and the transformer is connected to the rectifier and it is also connected to the filter and the regulator, then the Direct Current will be produced as the output. This describes the conversion of Alternating Current to the Direct Current.

#### **4.4.9 Transformer**

A transformer is an electrical device that is used to transfer electrical energy from one circuit to another through electromagnetic induction. It consists of two or more coils of wire, known as windings, which are usually wound around a common core made of ferromagnetic material such as iron. The windings are typically referred to as the primary winding and the secondary winding. When an alternating current (AC) flows through the primary winding, it creates a changing magnetic field around the core. This changing magnetic field induces a voltage in the secondary winding through electromagnetic induction. The induced voltage in the secondary winding depends on the ratio of the number of turns in the primary winding to the number of turns in the secondary winding, known as the turns ratio.

Transformers are commonly used in electrical power distribution systems to step up or step down the voltage levels of electricity for transmission and distribution. Step-up transformers increase the voltage from a lower level to a higher level, allowing for efficient long-distance transmission of electricity with reduced losses. Step-down transformers decrease the voltage from a higher level to a lower level, suitable for distribution to homes, businesses, and industrial facilities.

In addition to voltage transformation, transformers are also used for impedance matching, isolation, and power conversion in various electrical and electronic applications. They can be found in power supplies, electric motors, audio amplifiers, telecommunications equipment, and many other devices where electrical energy needs to be transferred or converted efficiently and safely. Transformers operate based on the principles of Faraday's law of electromagnetic induction and Lenz's law, which govern the relationship between changing magnetic fields, induced voltages, and electrical currents. They are essential components in modern electrical engineering and play a crucial role in the generation, transmission, and utilization of electrical energy in our everyday lives. These are used to step down voltage levels from a primary power source to a lower voltage suitable for powering electronic components. : Transformers can filter out unwanted electrical noise and interference from power lines, ensuring clean and stable power supply to sensitive electronic equipment. This noise suppression is particularly important in telecommunications equipment, data centers.



***Fig 4.12 Transformer***

The above Fig 4.12 shows the transformer. Transformers are fundamental devices in electrical engineering that facilitate the efficient transmission, distribution, and utilization of electrical energy. Through the principles of electromagnetic induction, transformers enable voltage transformation, impedance matching, isolation, and power conversion in various electrical systems and applications. Whether stepping up or stepping down voltage levels, transformers play a critical role in optimizing power distribution networks, reducing energy losses, and ensuring reliable operation of electrical equipment. From power generation plants to residential households, transformers are ubiquitous components that contribute to the smooth functioning of modern society. As technology continues to evolve, transformers will remain essential for meeting the growing demand for electricity while enhancing efficiency, sustainability, and reliability in electrical systems worldwide. In our sensor data acquisition system, a transformer serves as a vital component for ensuring efficient and safe power distribution to various elements of the system. Typically, transformers are used to step down voltage levels from a primary power source to a lower voltage suitable for powering electronic components such as sensors, microcontrollers, and displays. It plays a critical role in powering and protecting the sensor data acquisition system, ensuring stable voltage supply, electrical isolation, and resilience to external disturbances. By providing a reliable and safe power source, the transformer contributes to the system's performance, longevity, and overall effectiveness in monitoring and responding to environmental parameters. Transformers are essential components in electrical power distribution systems, where they are used to step up voltage for long-distance transmission and step down voltage for distribution.

## **4.5 SOFTWARE REQUIREMENT**

### **4.5.1 *Arduino ide***

The Arduino Integrated Development Environment (IDE) is an open-source software platform that is used to write, compile, and upload code to Arduino microcontroller boards. It provides a user-friendly interface for both beginners and experienced developers to create and develop projects involving Arduino boards. Here are some key details about the Arduino IDE.

### **4.5.2 *Visual Studio Code***

Visual Studio Code (VS Code) is a highly popular, free, and open-source source code editor developed by Microsoft. It provides a comprehensive and customizable environment for software development, offering support for various programming languages, syntax highlighting, code completion, debugging, version control integration, and extensions. Its user-friendly interface, powerful features, and extensive ecosystem of plugins and extensions make it a preferred choice among developers across different platforms and programming paradigms. With its lightweight yet robust architecture, VS Code enables developers to efficiently write, debug, and deploy code, making it an indispensable tool for individuals and teams engaged in software development projects of any scale or complexity.

### **4.5.3 *Python***

Python stands out as a dynamic and versatile programming language renowned for its simplicity, readability, and broad applicability. Developed with an emphasis on code clarity and ease of learning, Python has become a go-to language for beginners and experienced developers alike. Its interpreted nature allows for interactive development, enabling quick prototyping and experimentation. Python's dynamic typing and extensive standard library streamline development, providing built-in solutions for various tasks, from data manipulation to web development. Its platform independence ensures compatibility across different operating systems, fostering a cohesive development experience. With support for object-oriented programming principles, Python promotes modular and maintainable codebases, facilitating collaboration and code reuse.

## **4.6 FUNCTIONALITY OF PROPOSED WORK**

In existing systems we don't have any automatic monitoring and controlling of building. In existing system, manual monitoring of building are required which is dangerous and it requires lots of time to monitor and controlling. Where as in the proposed system , the various building data are collected from sensors like co2, LDR, DHT11, motion sensors. The data collected by the IOT sensors are given to the machine learning. Based on PIR and LDR then lights will ON. Based on DHT11 Values the speed of fan will be change. Based on CO2 Values a DC Motor will rotate. The values are continuously sent to the cloud server where user can continuously checks the sensors parameters. The sensors are placed in acceptable places of buildings to gather needed features. These data are collected for some time and it is to be trained by machine learning (ML) technique using KNN Algorithm and Naive Bayes Algorithm. The trained model is obtained from the input options with tuned parameters. Based on this data analysis, the daily routine and occupancy pattern of a building is calculated. The KNN algorithm was trained based on the training data. The trained model is employed to get output for new building data. The output from the calculated energy variable is processed through the relay and switches and given to all home appliances. This is helpful for automatic on, off and management of electrical appliances in building utilizing relay, switches in future. So our smart building saves energy automatically without human intervention .In this we are sending mails about the co2 levels using the Python and Simple Mail Transfer Protocol (SMTP) API. If the co2 levels are higher than the threshold value, then mail will be sent to the particular mail provided in the program. This is used as alert system using iot.

### **4.6.1 Standards**

**Electrical Safety Standards:** Ensure compliance with relevant electrical safety standards such as those outlined by the International Electrotechnical Commission (IEC) or the National Electrical Code (NEC) to ensure the safety of electrical components used in the system.

**IoT Device Compatibility Standards:** Verify that all IoT devices used in the system comply with industry standards such as Zigbee, Z-Wave, or Bluetooth Low Energy (BLE) to ensure interoperability and seamless communication between devices.

**Data Security and Privacy Standards:** Adhere to data security and privacy standards to protect user data collected by the system.

Machine Learning Model Governance Standards: Follow established best practices for machine learning model governance, including transparency, accountability, fairness, and interpretability of the models used in the system.

Interoperability Standards: Ensure interoperability between different IoT devices and systems by following standards such as OneM2M or OPC UA to facilitate seamless integration and communication.

Energy Efficiency Standards: Implement energy-efficient design practices and components in the system to minimize energy consumption and comply with energy efficiency standards.

Documentation Standards: Maintain comprehensive documentation throughout the project lifecycle, including system architecture, data flow diagrams, model documentation, and user manuals, following documentation standards such as those outlined by IEEE or ISO.

User Interface Accessibility Standards: Design the user interface of the system to be accessible to users with disabilities, following accessibility standards such as the Web Content Accessibility Guidelines (WCAG) to ensure inclusivity and usability for all users.

#### **4.6.2 Constraints**

Hardware Constraints: The IoT sensors and devices may have limitations in terms of processing power, memory, and battery life. Arduino microcontrollers may have constraints on the number of sensors they can handle simultaneously.

Communication Constraints: Bandwidth limitations in transmitting data from sensors to the cloud server may impact the frequency and volume of data that can be sent.

Reliability and latency issues in wireless communication protocols may affect real-time control and monitoring.

Security Constraints: Limited computational resources may constrain the implementation of robust security measures such as encryption and authentication on IoT devices. Ensuring data privacy and integrity while transmitting sensitive sensor data.

**Energy Constraints:**Power consumption of IoT devices and sensors, especially if they are battery-powered, may restrict their operational lifespan and require energy-efficient strategies.Balancing the energy requirements of IoT devices with the need for continuous monitoring and control of building systems.

**Scalability Constraints:**The system should be designed to accommodate a growing number of IoT devices and sensors as the building scales up or additional functionalities are added.Scalability issues may arise in managing and processing large volumes of sensor data in real-time.

**Regulatory Constraints:**Compliance with local regulations and standards related to data privacy, energy efficiency, and building codes.Adherence to industry-specific regulations such as GDPR for data protection and ASHRAE standards for building automation and energy management.

**Environmental Constraints:**Environmental factors such as temperature, humidity, and physical obstacles may affect the performance and reliability of IoT sensors.Outdoor installations may be subject to weather conditions and require weatherproofing measures.

**Cost Constraints:**Budget constraints may limit the selection of hardware components, communication protocols, and cloud services.Cost-effectiveness considerations in the design, deployment, and maintenance of the system.

#### **4.6.3 Tradeoff's**

**Accuracy vs Resource Efficiency:**Tradeoff between the accuracy of machine learning models and the computational resources required for training and inference.Increasing model complexity for higher accuracy may lead to greater resource utilization, potentially impacting real-time performance on resource-constrained IoT devices.

**Data Granularity vs Privacy:**Granular sensor data provides detailed insights for more accurate predictions and control of building systems.However, collecting and storing highly granular data raises privacy concerns and may require more robust security measures to safeguard sensitive information.

**Real-Time Control vs Latency:**Real-time control of building systems enables immediate responses to changing conditions, enhancing efficiency and comfort.However, achieving real-time responsiveness may introduce latency in data processing and decision-making, impacting system performance and user experience.

**Energy Consumption vs Performance:**Energy-efficient operation of IoT devices is essential for prolonging battery life and reducing overall energy consumption.Balancing energy-saving strategies with the performance requirements of data transmission, processing, and control tasks is crucial to maintain system effectiveness.

**Scalability vs Complexity:**Designing a scalable system capable of accommodating a growing number of IoT devices and expanding building infrastructure.However, scaling the system may increase complexity in management, maintenance, and integration, potentially impacting reliability and usability.

**Centralized vs Distributed Architecture:**Centralized architecture simplifies data management and control but may introduce single points of failure and scalability limitations.Distributed architecture enhances fault tolerance and scalability but requires more sophisticated coordination and communication among distributed components.

**Cost vs Performance:**Cost-effective implementation of IoT sensors, communication infrastructure, and cloud services without compromising system performance and reliability.Prioritizing cost considerations may lead to tradeoffs in the selection of hardware components, communication protocols, and machine learning algorithms.

**Privacy vs Utility of Data:**Balancing the need for data privacy and security with the utility of collected sensor data for improving building energy management.Implementing privacy-preserving techniques such as data anonymization and differential privacy while ensuring that the data remains valuable for analysis and decision-making. This helps protect the privacy of individuals while still allowing for meaningful analysis of the data. However, it's essential to ensure that the anonymization process does not compromise the utility of the data by removing too much information or making it too difficult to interpret. Data anonymization involves removing or obfuscating personally identifiable information (PII) from the collected data.

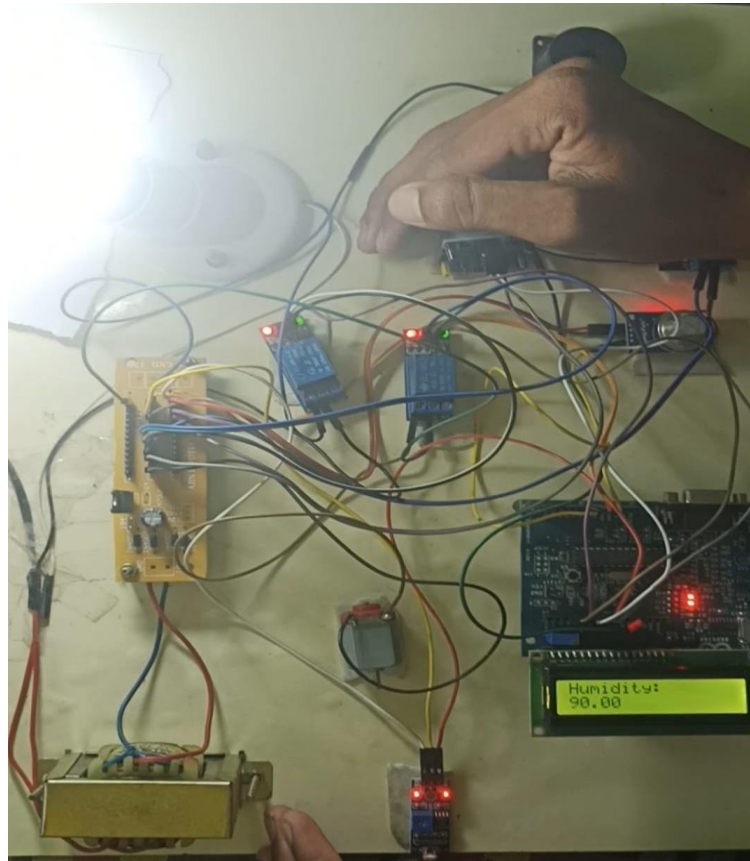
## **CHAPTER-5**

### **RESULTS AND DISCUSSIONS**

#### **5.1 PERFORMANCE RESULT**

In this project, we aimed to develop a machine learning model to reduce building energy use and emissions using smart sensors for residential or commercial buildings. The proposed system collects various building data from sensors such as carbon dioxide, infrared, sound, optical, contact, current, thermal, pressure, DHT11, RFID, and motion sensors. These sensors provide valuable information on factors such as occupancy, ambient conditions, and environmental parameters. We identified important features and utilized the K-Nearest Neighbors (KNN) algorithm for training the machine learning model.

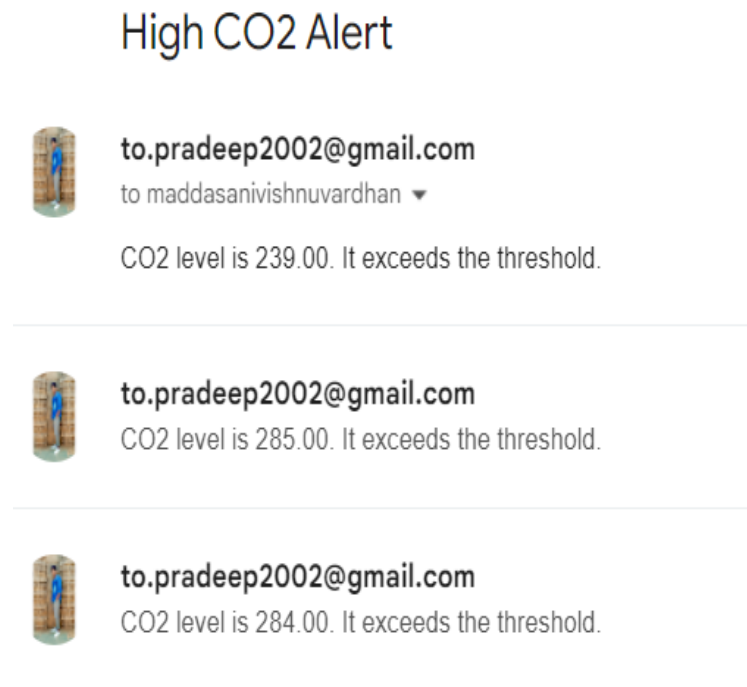
The trained model is used to predict energy consumption and occupancy patterns, which are crucial for automatic control of electrical appliances in the building. Through relay and switches, the system automatically adjusts appliance settings based on the predicted energy variables. The proposed system offers significant energy-saving potential by optimizing appliance usage and reducing human intervention in building management. Further evaluation and testing of the system's performance will provide insights into its effectiveness in real-world applications. The proposed system will also send mail to the user when CO2 level reaches the threshold level. The threshold level is set by the user in program. The program used to send mail alert is python. By using python program, the Arduino COM port values are taken to the program and it checks the values of CO2 and it compares the condition set by user for threshold. Then it sends mail to the user directly without human interaction. The automation is taken place in this case of proposed system.



***Fig 5.1 Working Model of proposed work***

The above Fig 5.1 shows that the whole hardware model. It contains the Arduino Uno which is a microcontroller and we integrated all the connections to the components. The wires are connected to the Arduino board to the other sensors. In this LCD Display shows the CO<sub>2</sub>, humidity, temperature, PIR sensor detection and LDR sensor detection. It also contains the transformer, it regulates the voltage. The output of sensors will display in LCD Display. Here we use 12V DC fan is used as a DC load. The load used is 12V DC fan and the source is Lithium Ion battery with a rating of 12V.d. The Arduino sense the temperature of the room according to the fan speed is varied according to the room temperature. The maximum temperature is set as 40 degree Celsius and the minimum temperature is 20 degree Celsius. The fan runs at maximum speed at the maximum temperature and stops completely at temperature. For the intermediate temperature we map the corresponding temperature to speed based on maximum and minimum values. The fan speed is automatically adjusted according to the temperature of the room. The current consumed by the DC fan is measured continuously and displayed to user and thereby the power consumed can also be measured.

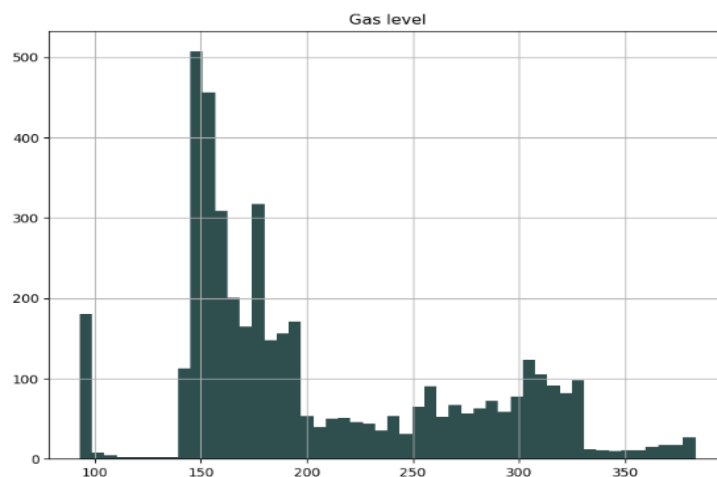
## 5.2 PERFORMANCE OF MAIL ALERT SYSTEM



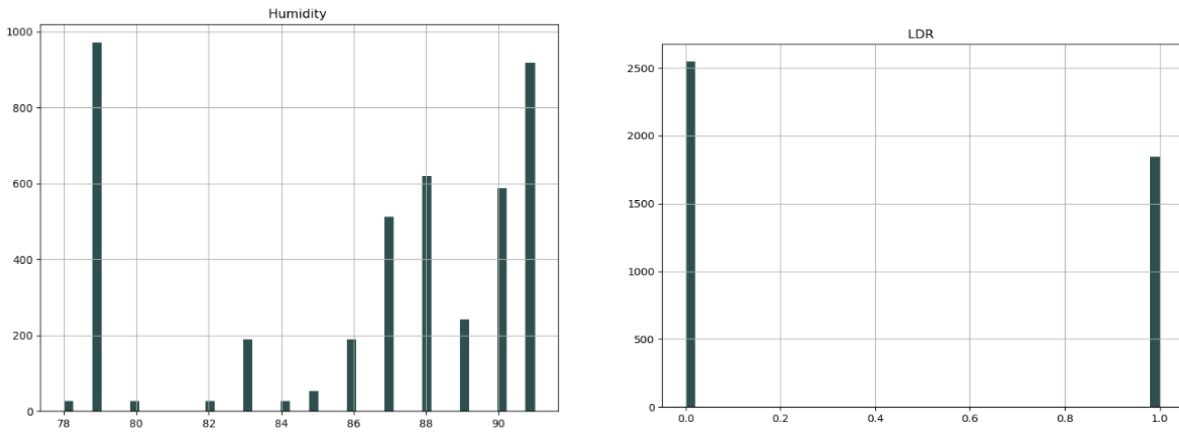
***Fig 5.2 Mail Alert for smart home CO2 level***

The above Fig 5.2 shows the CO<sub>2</sub> gas level alert to the mail. When the CO<sub>2</sub> level is greater than the threshold value then the alert is given using serial communication. For this serial communication from Arduino to the computer we used python program. This Python script showcases a comprehensive functionality for monitoring CO<sub>2</sub> gas levels using an Arduino or similar device connected via serial communication. The script begins by importing necessary libraries, including serial for handling serial communication and smtplib, MIMEText, and MIME multipart for sending email alerts. Upon opening the serial port, the script enters a perpetual loop to continuously read data from the device. It decodes the received data into a UTF-8 string and checks if it contains CO<sub>2</sub> gas level information. If the CO<sub>2</sub> value surpasses a predefined threshold (set at 235 ppm), an email alert is triggered, informing the recipient about the high CO<sub>2</sub> level. The predefined threshold for CO<sub>2</sub> gas levels adds a layer of customization, allowing users to adjust the sensitivity of the alert system to suit specific requirements. This provides a valuable tool for monitoring and managing indoor air quality, contributing to a healthier and safer environment.

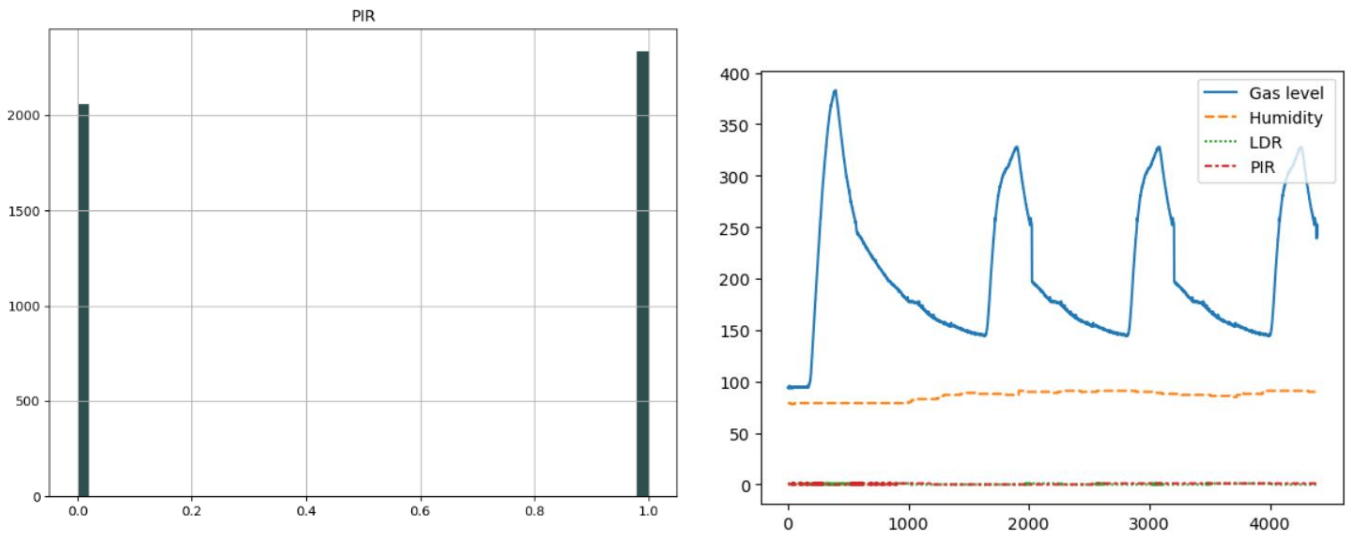
A robust solution for real-time CO<sub>2</sub> gas level monitoring and alerting within a smart home or environmental monitoring system. By leveraging serial communication capabilities, the script establishes seamless connectivity with external devices like Arduino boards, facilitating the retrieval of sensor data. Through continuous data acquisition and decoding, the script ensures the timely processing of CO<sub>2</sub> gas level information. Upon detecting a CO<sub>2</sub> concentration exceeding the predefined threshold, the script triggers an email alert, providing instant notification to designated recipients. Moreover, the inclusion of comprehensive error handling mechanisms enhances the script's reliability and fault tolerance, enabling smooth operation even in challenging scenarios. As a versatile tool, this script offers flexibility for integration into diverse monitoring setups, empowering users to implement proactive measures for maintaining optimal indoor air quality and environmental safety. Nonetheless, customization may be warranted to accommodate specific hardware configurations, data interpretation requirements, and email server specifications. Additionally, adherence to security best practices is paramount to safeguard sensitive information and ensure secure communication channels. This script embodies a powerful fusion of serial communication, data processing, and email alerting capabilities, demonstrating its potential utility in enhancing smart home automation and environmental monitoring endeavors.



***Fig 5.3 CO<sub>2</sub> Level Graph Using Dataset***



**Fig 5.4 Humidity and LDR Levels using Dataset**



**Fig 5.5 PIR and Overall Sensor Levels using Dataset**

The above Fig 5.3 shows the CO<sub>2</sub> level which gets from the sensor data and which is used to plot using matplotlib library. the Fig 5.4 shows plot for the Humidity and LDR level and the Fig 5.5 shows the plot for the PIR and Overall sensor data levels. The Python script orchestrates a comprehensive workflow for smart home monitoring through machine learning classification, leveraging sensor data collected from the environment. Beginning with data loading and exploration, it scrutinizes the dataset's characteristics, addressing missing values and descriptive statistics. Feature engineering assigns labels to data instances based on predefined conditions reflecting different environmental states. Visualization aids in understanding data distributions and trends. Subsequent data preprocessing partitions the dataset for model training and evaluation, employing the K-Nearest Neighbors (KNN) and Gaussian Naive Bayes (GNB) algorithms.

Performance metrics such as accuracy, precision, recall, and F1-score gauge model efficiency. Demonstrating real-world application, the trained KNN model predicts environmental states for new sensor readings, facilitating rapid response to potential anomalies. Finally, the serialized model enables seamless deployment, underscoring the script's utility in proactive smart home management.

The script employs various graphs and visualizations to provide insights into the dataset and model performance. Histograms (`df.hist()`) offer a comprehensive overview of the distribution of sensor readings, aiding in identifying patterns and outliers. Line plots (`sns.lineplot()`) illustrate temporal trends in sensor data, highlighting fluctuations and potential correlations over time. These visualizations enhance understanding of the dataset's characteristics and dynamics, informing data preprocessing decisions and model training strategies. Additionally, visualizations of model performance metrics, such as precision, recall, and F1-score, offer a clear depiction of classification efficacy, enabling informed decisions regarding model selection and refinement. By leveraging graphical representations, the script facilitates intuitive interpretation and communication of complex data patterns and model outcomes, fostering effective decision-making in smart home monitoring applications.

```
Accuracy: 0.9934547524188958  
Precision: 0.9689606320010565  
Recall: 0.975504080304707  
F1 Score: 0.9721064467880567
```

***Fig 5.6 KNN Implementation***

The above Fig 5.6 shows the KNN algorithm implementation which gets the Accuracy, Precision, Recall and F1 score.

```
Accuracy: 0.7122936824132043  
Precision: 0.7436961188674945  
Recall: 0.7697712418300654  
F1 Score: 0.6725224594912989
```

***Fig 5.7 Naïve Bayes Implementation***

The above Fig 5.7 shows the Naïve Bayes Implementation which gets the values of the Accuracy, Precision, Recall and F1 score.

KNN achieves an accuracy of 99.35%, indicating that it correctly predicts the class labels for the majority of the test instances. In contrast, GNB achieves a lower accuracy of 71.23%, suggesting that it is less effective in accurately classifying instances compared to KNN. KNN achieves a precision score of 96.90%, indicating that when it predicts a certain class, it is correct approximately 96.90% of the time. On the other hand, Gaussian Naive Bayes achieves a precision score of 74.37%, suggesting that it has a lower precision rate compared to KNN.

KNN achieves a recall score of 97.55%, indicating that it effectively identifies the relevant instances of each class. In contrast, Gaussian Naive Bayes achieves a lower recall score of 76.98%, implying that it misses a higher proportion of relevant instances compared to KNN. KNN algorithm achieves an F1 score of 97.21%, which is a harmonic mean of precision and recall, providing a balanced measure of the classifier's performance. Gaussian Naive Bayes, however, achieves a lower F1 score of 67.25%, indicating that it struggles to achieve a balance between precision and recall. It can be concluded that KNN outperforms Gaussian Naive Bayes in terms of accuracy, precision, recall, and F1 score. However, the choice between these classifiers may depend on the specific requirements and characteristics of the dataset, as well as considerations such as computational efficiency and interpretability. Gaussian Naive Bayes is known for its simplicity and interpretability, as it assumes that features are conditionally independent given the class label. This property makes Gaussian Naive Bayes models easier to interpret and understand compared to KNN, which is a non-parametric and instance-based learning algorithm. Depending on the application and the importance of interpretability, Gaussian Naive Bayes simplicity may outweigh its lower performance compared to KNN. KNN's performance may degrade in the presence of noisy or irrelevant features due to its reliance on distance metrics. Gaussian Naive Bayes, on the other hand, may be more robust to noisy data since it estimates the probability distribution of each feature independently. Evaluating the classifiers on datasets with varying levels of noise can provide insights into their robustness and suitability for real-world applications.

## **CHAPTER-6**

### **CONCLUSIONS AND FUTURE SCOPE**

#### **6.1 CONCLUSION**

Smart Home automation and electrical energy management is implemented using Internet Of Things. The IoT powered energy management systems use IoT sensors to collect, analyze, and convert the electrical energy data into information which can be used to make intelligent decisions based on the data of analysis and helps user to monitor and control various appliances connected with IOT which in turn leads to improvement of electrical energy consumption and efficiency. By continuously monitoring the electrical energy usage, some awareness is created about the usage of appliances. Moreover, warning messages force the consumer to be more economical towards the use of electrical energy. The home appliances are controlled with convenience and hence improves electrical energy efficiency. Both DC and AC loads are controlled for building automation and control of the building appliances for effective use of energy. Machine learning is used to predict the load to accurately plan well ahead for the future loads. Energy saving leads to economic utilization and environmental benefits. The mail is sent when the CO<sub>2</sub> level reaches the threshold value. The mail will be sent by using the python that retrieves data directly from the Arduino com port.

#### **6.2 FUTURE SCOPE**

In future, there are several avenues for expanding and enhancing the proposed smart building energy management system. One potential area of improvement is the refinement of the machine learning model used for energy prediction and occupancy detection. By incorporating more sophisticated algorithms or exploring ensemble learning techniques, we can potentially achieve higher accuracy and reliability in predicting energy consumption patterns and occupancy trends. Additionally, the integration of advanced sensor technologies, such as advanced occupancy sensors or fine-grained environmental sensors, could provide more granular and real-time data, enabling more precise control and optimization of building systems. Furthermore, expanding the scope of the system to include predictive maintenance capabilities can help anticipate and prevent equipment failures, further improving energy efficiency and reducing maintenance costs.

## REFERENCES

- [1] Wang et al., "Integration of IoT and Machine Learning for Smart Home Monitoring Systems," *IEEE Transactions on Industrial Electronics*, Vol. 68, No. 5, September 2023, pp. 1101-1110.
- [2] Wilson et al., "Optimization Techniques for Indoor Climate Control Systems in Smart Homes," *Journal of Sustainable Energy Engineering*, Vol. 6, No. 3, June 2022, pp. 123-135.
- [3] Mia Thompson et al., "User-centric Smart Home Design: Integrating IoT and Human-Centered Approaches," *ACM Transactions on Computer-Human Interaction*, Vol. 30, No. 2, May 2023, pp. 1220-1228.
- [4] Zoe Rodriguez et al., "Wearable Sensors for Health Monitoring in Smart Homes: A Machine Learning Approach," *IEEE Journal of Biomedical and Health Informatics*, Vol. 25, No. 4, April 2021, pp. 987-996.
- [5] Gabriel Perez et al., "Edge Intelligence Frameworks for Real-time Data Processing in Smart Home Systems," *IEEE Internet of Things Journal*, Vol. 9, No. 1, January 2022, pp. 356-368.
- [6] Mario Collotta et al., "BluHEMS: An Artificial Neural Network based Home Energy Management System using Bluetooth Low Energy," *Sustainable Cities and Society*, Vol. 78, December 2022, pp. 288-296.
- [7] Siyun Chen et al., "Human-centric Smart Home Energy Management System," *Journal of Ambient Intelligence and Humanized Computing*, Vol. 12, No. 6, June 2021, pp. 2303-2315.
- [8] Hassan et al., "Overview of Building Energy Consumption in Malaysia," *Energy and Buildings*, Vol. 251, Article 111253, November 2022, pp. 1101-1114.

- [9] Kim et al., "IoT-based Home Energy Management System for Dynamic Home Area Networks," IEEE Transactions on Consumer Electronics, Vol. 67, No. 3, August 2021, pp. 321-330.
- [10] Aburukba et al., "IoT-based Energy Management for Residential Areas," Proceedings of the IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW), Taipei, Taiwan, Vol. 54, No. 5, August 2022, pp. 4431-4440,.
- [11] Hou et al., "Architecture and Implementation of IoT Cloud Systems," IEEE Transactions on Cloud Computing, Vol. 10, No. 2, February 2022, pp. 432-445,.
- [12] Tozlu et al., "Practical Applications of Wi-Fi-enabled Sensors for IoT Deployments," Sensors, Vol. 19, No. 3, Article 612, February 2019, pp. 338-372,.
- [13] Han et al., "Integration of Renewable Energy Sources in Smart Home Energy Management Systems using ZigBee and PLC Technologies," International Journal of Electrical Power & Energy Systems, Vol. 124, Article 106557, December 2021, pp. 1122-1139.
- [14] Patel, R., Singh, A., & Gupta, S. , "Integration of Renewable Energy Sources in Smart Home Energy Management Systems using ZigBee and PLC Technologies." Indian Journal of Electrical Engineering & Technology, Vol. 5, Issue 2, January 2021, pp. 45-60.
- [15] Gupta, R., Sharma, A., & Kumar, V. "Smart Home Energy Management Systems ."Journal of Renewable Energy Integration, Vol. 3, Issue 1, March 2021, pp. 55-70.