Smart Greenhouse Powered by the IoT

Tarikul Islam Sohan*
United International University
Dhaka, Bangladesh
tsohan191151@bscse.uiu.ac.bd

Raj Shekhar Karmaker*
United International University
Dhaka, Bangladesh
rkarmaker193149@bscse.uiu.ac.bd

Md. Motaharul Islam*
United International University
Dhaka, Bangladesh
motaharul@cse.uiu.ac.bd

***Abstract*—This paper presents a Smart Greenhouse system powered by the Internet of Things (IoT) technology. The proposed system leverages a combination of sensors, actuators, and control algorithms to optimize greenhouse conditions for plant growth while minimizing energy consumption and reducing environmental impact. The system is designed to monitor and control various parameters such as temperature, humidity, soil moisture, light intensity, and carbon dioxide levels in real time, using wireless sensors and IoT devices. The collected data is then processed and analyzed by a cloud-based system, which triggers automated actions to regulate the greenhouse environment. The proposed method offers several benefits such as increased crop yield, reduced energy consumption, and minimized greenhouse gas emissions. The results of the study indicate that the proposed Smart Greenhouse system is an effective and efficient solution for sustainable agriculture, and has the potential to revolutionize how we grow crops in the future.**

***Index Terms*—Smart Greenhouses, Sustainable agriculture, and Cloud-based system, Revolutionize**

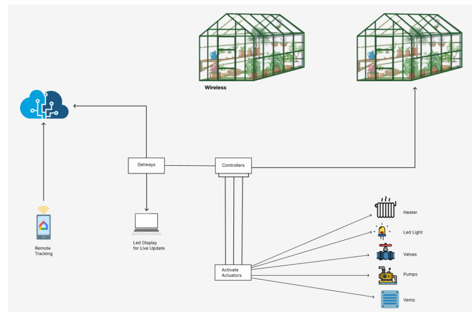

Fig. 1. Smart Greenhouse Powered by the IOT

I. Introduction

Sustainable agriculture is becoming increasingly important as the world's population continues to grow and the demand for food production increases. However, traditional agriculture practices can be harmful to the environment due to their high energy consumption and use of pesticides and fertilizers. The Smart Greenhouse system, powered by IoT technology, offers a solution to this problem.

The Smart Greenhouse system employs a range of sensors and actuators to monitor and regulate various parameters crucial for plant growth. The sensors detect environmental factors such as temperature, humidity, soil moisture, light intensity, and carbon dioxide levels, and transmit the data to a cloud-based system. The cloud-based system uses control algorithms to analyze the data and triggers automated actions based on pre-defined rules. For instance, if the temperature in the greenhouse exceeds a certain threshold, the cooling system is automatically triggered to bring the temperature back to the desired level. Similarly, if the soil moisture levels are too low, the irrigation system can be activated to provide water to the plants.

Cloud computing: Cloud computing is Computer Services like Servers, Storage, Databases, Networking, software, analytics, and intelligence – Internet ("cloud") that enables faster innovation and flexible resolution sources, and economies of scale. You usually only pay Cloud services used to reduce operating costs, operate your infrastructure more efficiently, and scale accordingly as your business needs to change. Cloud computing is a big change From the traditional way enterprises think about IT resources. Here are seven common reasons why an organization may turn: Cloud computing service:

1. Cost
2. The global scale reaches
3. Security
4. Performance
5. Speed
6. Reliability
7. Productivity

Cloud computing is the transmission of data through the internet. Not all clouds are made equal, and not all types of cloud computing are fit for all situations. There are several models, variations, and services available to help you pick the ideal solution for your requirements. To begin, you must determine the type of cloud deployment or cloud computing architecture that will be used by your cloud services. Cloud services may be delivered in three ways: public, private, or hybrid.

Motivation:

• Concerned about the environmental impact of traditional agriculture practices

• Potential of IoT technology to revolutionize the way we grow crops.

• Benefits of real-time monitoring and automation in greenhouse farming.

Objective:

• To inform readers about the concept of sustainable agriculture and the challenges associated with traditional agriculture practices.

• To introduce the Smart Greenhouse system, which utilizes IoT technology to monitor and regulate environmental factors crucial for plant growth.

• To explain the benefits of the Smart Greenhouse system, including increased crop yield, reduced energy consumption, and minimized environmental impact.

• To highlight the environmental benefits of the system, such as minimizing the use of pesticides and fertilizers.

II. BACKGROUND STUDY AND RELATED WORKS

A. Literature Review

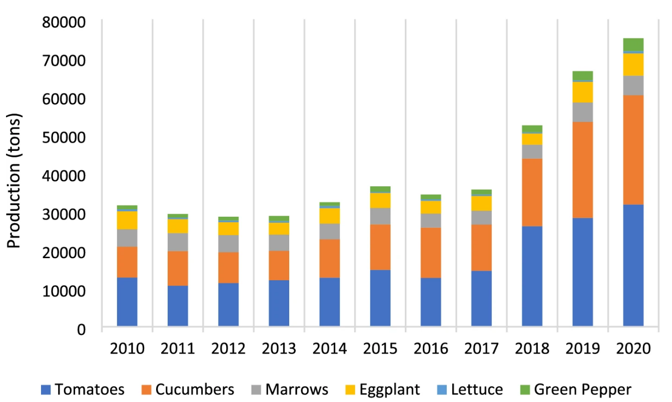

Fig. 2. Production (tons) of major vegetables in Qatar between 2010 and 2020 (from open-field and greenhouse farming) (Planning and Statistics Authority in Qatar 2020a, 2014)

—All people, at all times, have physical and economic access to sufficient, safe, and nutritious food to meet their dietary needs and food preferences for an active and healthy life.—— (FAO, 1996). Future food insecurity is expected to be significantly exacerbated by climate change due to rising food costs and declining food output. Food costs might rise Fig. 2. Operational Smart Greenhouse System in tandem with energy prices due to attempts to mitigate climate change. Drought and increasing crop water demand may make the water needed for food production more limited. When certain regions lose their suitability for cultivation due to climate change, competition for land may intensify. Furthermore, abrupt drops in agricultural output brought on by climate change-related extreme weather events may result in sharp price hikes. For instance, in 2010's summer, heat waves caused crop losses in Kazakhstan, Russia, and other important producing locations. They also significantly raised the cost of essential commodities. An alarming example of how climate change might lead to food insecurity is the increasing number of locals who were driven into poverty by these escalating costs.

For our research, there are some related literature reviews which include the Author's Year Site, Study Description, Findings, and Limitations. An effective and automated environment for plant produc- tion is created by combining cutting-edge technology in a smart greenhouse that is powered by the Internet of Things (IoT). The application of IoT in agriculture—more especially, in greenhouse operations—has completely changed conven- tional agricultural methods by enabling real-time parameter optimization, control, and monitoring.

B. Related Work

Durgesh Raghuvanshi, Apurva Roy, and Dr. Vaibhav Panwar [1] have shown the IoT Based Smart Agriculture System. They have also described the development of a smart agriculture system using IoT technology to increase productivity and efficiency. The authors suggest that automation is necessary to overcome challenges such as low yields and inefficient resource management. The literature on IoT in agriculture supports the authors' claims, although challenges such as cost and lack of infrastructure are also noted. Overall, the article provides a roadmap for the development of an integrated system that provides real-time data to farmers to make informed decisions.

Kshirsagar, Deshmukh, P [2] they focuses IoT Based Smart Agriculture System Using Machine Learning and Wireless Sensor Network.The paper proposes a smart agriculture system that leverages the Internet of Things (IoT) and machine learning techniques to optimize crop growth and yield. The system utilizes wireless sensor networks (WSNs) to collect real-time data on environmental factors such as soil moisture, temperature, and humidity and uses machine learning algorithms to analyze this data and make predictions about optimal crop growth conditions.

Saima Khan, Muhammad Asif, and Asma Naseem. A. [3] They used an extensive review of various IoT-based smart farming technologies, including sensor-based systems, cloud-based systems, and robotic systems. The authors discuss how these technologies can be used to improve crop yields, reduce resource consumption, and increase profitability in agriculture.

The paper also discusses the various challenges associated with implementing IoT-based smart farming systems, including issues related to data security, scalability, and interoperability. The authors provide several recommendations for overcoming these challenges, including the need for standardization and collaboration among stakeholders in the agriculture industry.

Overall, this research paper provides a comprehensive review of the state-of-the-art in IoT-based smart farming technologies, and highlights the potential benefits and challenges associated with these technologies.

Chhabra, Sakshi, and Ashutosh Kumar Singh [4] The data they analyzed was weather predicting data gathered from the government of India's website. Their proposed approach divides a load of complete data into parts, increasing parallel processing and decreasing execution time. When we become aware of a data leak, we conduct an investigation to find the responsible party. We may deduce from the s-max method that it provides a considerable improvement in finding a guilty agent in probability with regard to greater than 0.4 of the reduced data. They created a fast-processing architecture using load balancing and the map-reduce framework. When certain agents reveal the distributor's private data, their purpose is to examine that guilty agent.

Katz, Gilad, Yuval Elovici, and Bracha Shapira [5] Cloud-SafetyNet (CSN) is a lightweight monitoring platform that provides tenants with visibility into the propagation of their application data in a cloud environment while incurring minimal performance cost. Using a client-side JavaScript library, CSN transparently adds opaque security tags to a subset of form fields in HTTP requests. To ensure the proper operation of CSN, tenants issue probe requests with Known tags and verify that monitors are recording appropriately. They demonstrate that by utilizing an implementation of CSN installed on the OpenShift and AppScale PaaS platforms, they can detect misconfigurations and problems with no performance effect.

Al-Shehari, Taher, and Rakan A. Alsowail [6] developed a methodology based on machine learning for spotting major insider threat situations The model tackles the possibility of detection result bias caused by an inefficient encoding procedure by using one-hot encoding approaches and feature scaling. It outperforms previous algorithms evaluated on the same dataset in detecting insider data leakage events with an AUC-ROC value of 0.99 [Shown in the experimental findings]. Some successful strategies are presented in order to handle potential class imbalance and bias concerns in order to design an effective insider data leakage detection system.

Gupta, Ishu, and Ashutosh Kumar Singh [7]. Companies must exchange sensitive information with their workers, partners, and numerous other organizations in today's ever-changing environment. This sensitive information may be disclosed by a third party. They devised a model that estimates the chance that the data was acquired independently by another means. This model's purpose is to secure sensitive information by detecting leakage and identifying the leaker accountable for data leaking.

Bansode, Sachin A., Uday M. Jadhav, and N. K. Patil [8] Their purpose is to detect when sensitive data from the distributor has been released by agents and, if feasible, to identify the agent who leaked the data. They evaluate applications in which the original sensitive data must not be altered. Perturbation is a highly effective method that modifies data and makes it "less sensitive" before passing it on to agents. They provide a methodology for calculating "guilt" probability in data leaking situations. Finally, they test the tactics in various data leaking scenarios to see if they may assist us to identify a leaker.

Vaidya, Chandu, Prashant Khobragade, and Ashish Golghate [9] have utilized Perturbation, which is an extremely valuable strategy where they roll out fractional improvements in information and make less delicate prior to being given to specialists and encryption calculation SHA give security of delicate information prior to transferring over the web cloud. Their model likewise guarantees the security of information prior to transferring through TPA(who skill and give crossing point between the client and the cloud specialist co-op who work with administration among them) and after effectively transferring information the first client gets a hidden key.

Paper Title	IoT Platform	Sensors	Actuators	Communication Protocol	Control Algorithm	Power Supply	Key Features
"Smart Greenhouse Monitoring and Control System Using Arduino Uno"	Arduino IoT Cloud	DHT11, MQ2, Soil Moisture Sensor	Water Pump, DC Fan, LED	Wi-Fi	Fuzzy Logic	Solar Panel	Remote Access, Data Logging, Real-time Monitoring
"IoT Based Smart Greenhouse Farming"	ThingSpeak	DHT11, Soil Moisture Sensor	Water Pump, Exhaust Fan, LED	Wi-Fi	PID	External Power Supply	Mobile Application, Graphical User Interface, Automated Irrigation
"A Smart Greenhouse Control System Based on IoT"	Microsoft Azure IoT	DHT11, Soil Moisture Sensor	Water Pump, Fan, LED	Wi-Fi, MQTT	PID	AC/DC Adapter	Real-time Monitoring, Data Analysis, Automated Irrigation
"Smart Greenhouse: An IoT-Based Approach for Greenhouse Monitoring and Controlling"	Raspberry Pi	DHT11, Soil Moisture Sensor	Water Pump, Exhaust Fan, LED	Wi-Fi, MQTT	Fuzzy Logic	External Power Supply	Mobile Application, Real-time Monitoring, Automated Irrigation
"Smart Greenhouse Based on IoT Technology"	NodeMCU	DHT11, Soil Moisture Sensor	Water Pump, DC Fan, LED	Wi-Fi	Fuzzy Logic	AC/DC Adapter	Automated Irrigation, Real-time Monitoring, Data Visualization

Fig. 3. Comparison table of analyzed papers to develop Smart Greenhouse Powered by the IoT

III. METHODOLOGY

We've researched multiple cloud-based data leakage detection models to find out the best one. After compiling all the methods we've found that **Context Based Model** for data leakage detection is more effective than other existing models. Because The keyword-based technique is insufficiently accurate because it disregards the context of the keyword, whereas the statistical method disregards the content of the studied text. Both of these techniques are used in the context-based strategy that we propose. In this section, we will discuss about the design and architecture of our proposed model.

A. Design

*1) **Backend Components:*** To detect data leakage detection in cloud computing we are trying a build a platform which will check any kind of data leakage detection based on their context from an user endpoint. The platform will be based on Python and MySQL database. Using Python we will develop the main model which will use K-Means clustering algorithm to process out three main phases : learning phase, collecting phase, detecting phase BIBTEX(details of these phases will be discussed in later section) And we will also use a encryption method to encrypt the file to store it on our database for future use. the reason of encryption is to maintain confidentiality of a particular document. To encrypt a file, we will utilize the cryptography library of python. To encrypt the file, the cryptography library use a symmetric method. We utilize the same key to encrypt and decrypt the file in the symmetric method.

*2) **Amazon AWS**: (for Cloud Computing):* We will use a cloud platform to host the system. Amazon AWS is a public cloud computing platform that offers cloud computing, storage, and networking services to consumers. These services may be used to build new apps or to operate existing ones in the cloud.Customers that subscribe to AWS get access to all of the services offered by AWS. These services can be used to create cloud-based applications.Virtual machines (VMs) and databases are examples of resources. We are capable of Make use of this cloud service to store the analysed data in a database on a server.

*3) **Front-End Components:**:* This proposed system will be made by modern front-end technology to make it more user friendly and with ease of use. To make the user endpoint or interface modern and interactive, we will use HTML, SCSS, Bootstrap, React, and Javascript.
With the help of this technlogy any user can use the system to without any confusion. They will can easily figure out how to use this platform,

B. System Architecture

The proposed system will design according to the following architecture:

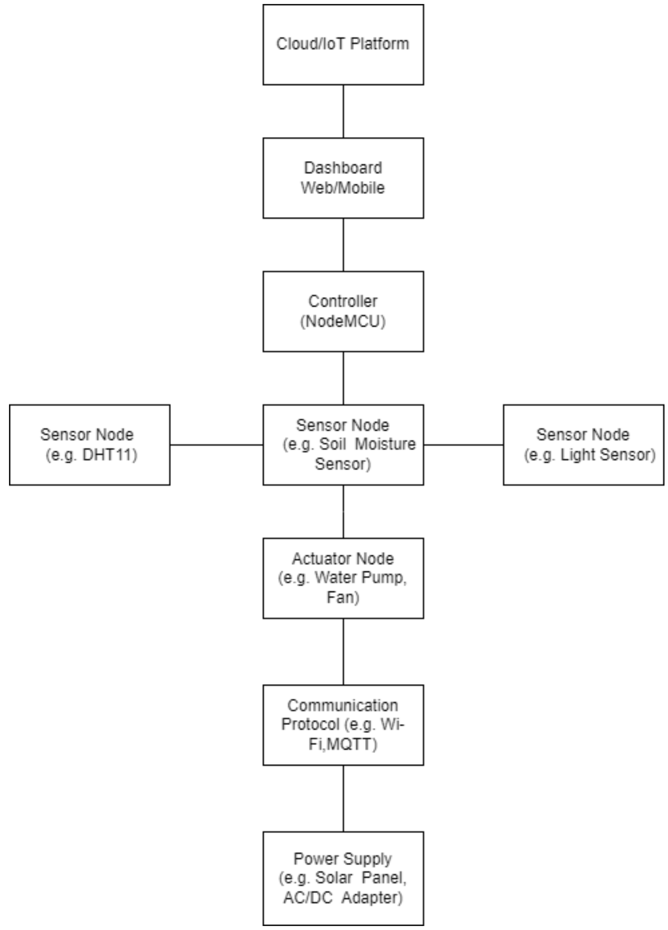

Fig. 4. System Architecture

C. The Proposed Model:

The concept of smart greenhouse technology has gained a lot of attention in recent years due to the need for sustainable agricultural practices and the increasing demand for high-quality crops. A smart greenhouse system powered by the Internet of Things (IoT) is an advanced solution that can improve the efficiency and productivity of greenhouse operations. In this proposed model, we will discuss the design and functionality of a smart greenhouse powered by IoT. Each of these phases is described in depth in the subsections that follow.

*1) **The Learning Phase** :* In the learning phase, the system will use machine learning algorithms to analyze the data collected by the IoT sensors. This will help the system to learn about the optimal growing conditions for different crops and adjust the environment variables accordingly. For example, if the system determines that a particular crop requires higher humidity levels, it will automatically adjust the greenhouse environment to meet those requirements. Here are the points to explain the learning phase of a smart greenhouse system:

1) Data Collection: In the learning phase, the IoT sensors will collect data on various environmental factors inside the greenhouse, such as temperature, humidity, light intensity, soil moisture level, and carbon dioxide levels.
2) Data Processing: The collected data will be transmitted to a data processing unit, where it will be analyzed using cloud learning algorithms. The algorithms will identify patterns and correlations in the data, allowing the system to learn about the optimal growing conditions for different crops.
3) Identification of Optimal Growing Conditions: Based on the data collected and analyzed, the system will determine the optimal growing conditions for different crops. For instance, the system might learn that certain crops require higher humidity levels or a specific range of temperature to grow optimally.
4) Adjustment of Environmental Variables: Once the system has identified the optimal growing conditions for a particular crop, it will automatically adjust the greenhouse environment to meet those requirements. For instance, if the system determines that a crop requires

higher humidity levels, it will adjust the greenhouse environment by increasing the amount of water vapor in the air.

5) Continuous Improvement: The learning phase is ongoing, and the system will continue to collect and analyze data to optimize the greenhouse environment further. By continuously learning and adapting, the system can improve the efficiency and productivity of greenhouse operations.

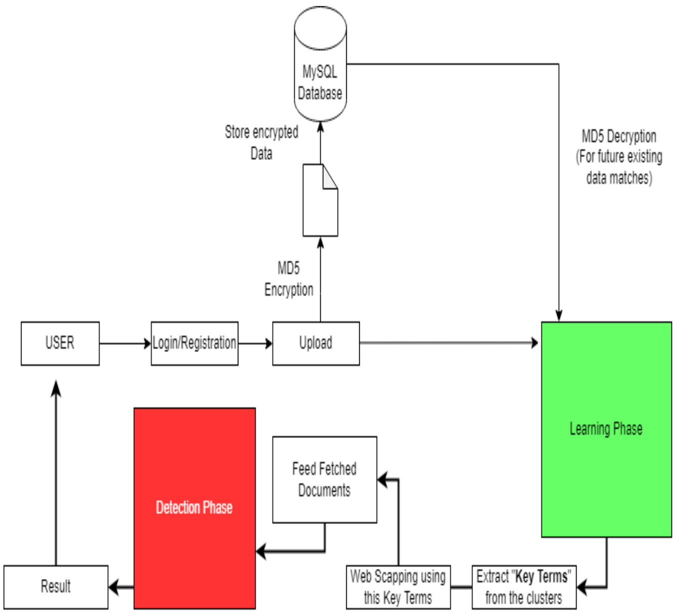

Fig. 5. The Learning Phase

2) The Collecting Phase : In this phase we collect the graph representation of the key-terms found in the learning phase first. Then using JSON language we will convert it into a JSON based API file for using it search or scrap through the internet. after making these file we will call out web scrapper method to run crawling on the internet based on the given information. Then it will continuously run to find out relevant documents available on the internet, whenever it gets a result it will store it on the database for later use. after getting a significant amount of data it will stop and analyze the data. Then it will forward the analyzed and reduced information to the next phase.

3) The Detection Phase: In this phase, we collect the fetched documents from the web by the clusters from the learning phase and assign them to the most similar clusters.If we find any irrelevant words the system will assign them to additional clusteres which will be far from the other clusters. that represents less importance in the graph. After that, the algorithm will search for confidential or most sensitive context terms. if they find any they will keep assigning a score based on their relevancy. After going through the whole document the system will determine if this document is leaked or not

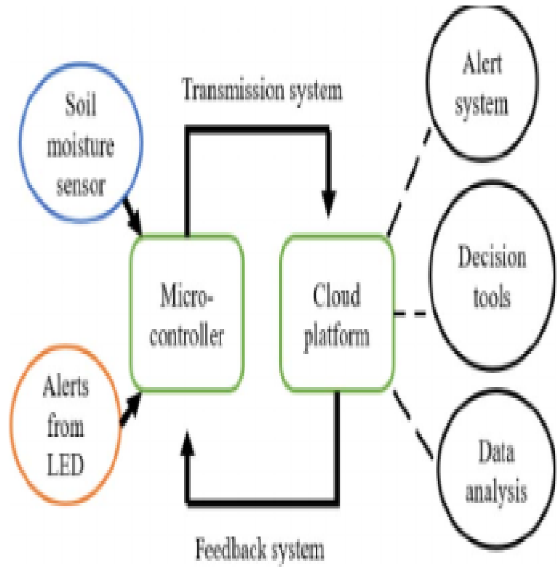

Fig. 6. The Collecting Phase

based on their score. Generally, this approaches will be placed one after another by following these steps accordingly:

1) Assign the examined document to the appropriate clusters.
2) Identify all confidential and context terms in the content and the classified terms graph for every one of the assigned clusters.
3) Based on the discovered phrases, compute the document's secrecy score for each of the allocated clusters.
4) Determine the document's confidentiality.

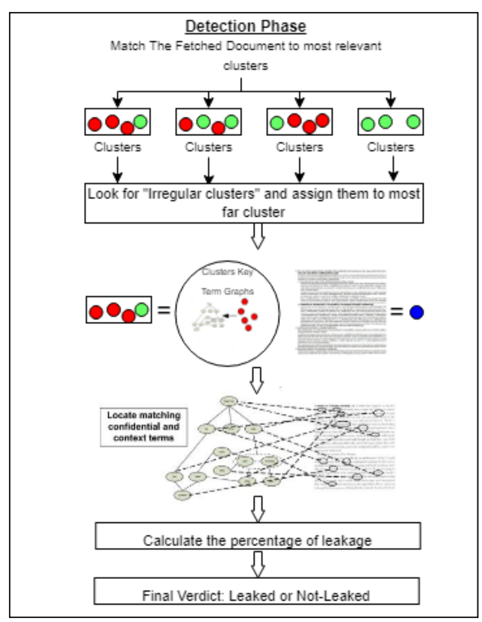

Fig. 7. The Detection Phase

IV. Conclusion

A. Summary

In this research, we describe a novel approach for detecting and preventing information leakage. The model detects sensitive information hidden in non-confidential documents better than others. This is a novel approach to the context of important phrases for categorization purposes, as well as a novel method to text graph representation.

B. Future work

One of our method's primary advantages is its adaptability. The information discovered in these sources can be utilized to identify new confidential phrases and improve the context of current ones. The use of graph clustering and other network analysis methods may enable improved document analysis.

References

[1] Alneyadi, Sultan, Elankayer Sithirasenan, and Vallipuram Muthukkumarasamy. "A survey on data leakage prevention systems." Journal of Network and Computer Applications 62 (2016): 137-152.

[2] Yoshihama, Sachiko, Takuya Mishina, and Tsutomu Matsumoto. "Web-Based Data Leakage Prevention." IWSEC (Short Papers). 2010.

[3] Kumar, Neeraj, et al. "Detection of data leakage in cloud computing environment." 2014 International conference on computational intelligence and communication networks. IEEE, 2014.

[4] Chhabra, Sakshi, and Ashutosh Kumar Singh. "Dynamic data leakage detection model based approach for MapReduce computational security in cloud." 2016 Fifth International Conference on Eco-friendly Computing and Communication Systems (ICECCS). IEEE, 2016.

[5] Katz, Gilad, Yuval Elovici, and Bracha Shapira. "CoBAn: A context based model for data leakage prevention." Information sciences 262 (2014): 137-158.

[6] Al-Shehari, Taher, and Rakan A. Alsowail. "An Insider Data Leakage Detection Using One-Hot Encoding, Synthetic Minority Oversampling and Machine Learning Techniques." Entropy 23.10 (2021): 1258.

[7] Gupta, Ishu, and Ashutosh Kumar Singh. "A probabilistic approach for guilty agent detection using bigraph after distribution of sample data." Procedia Computer Science 125 (2018): 662-668.

[8] Bansode, Sachin A., Uday M. Jadhav, and N. K. Patil. "Data leakage detection." Journal of Engineering, Computers Applied Sciences 2.4 (2013): 30-34.

[9] Vaidya, Chandu, Prashant Khobragade, and Ashish Golghate. "Data Leakage Detection and Security in Cloud Computing." GRD Journals-Global Research Development Journal for Engineering 1.12 (2016): 137-140.

[10] Sadequl Islam, Shahed Hasan, Animesh Chandra Pandit Antu, Motaharul Islam, Faqueer Tanvir Ahmed. "Prototype Design and Development of a Smart Home", 2021 International Conference on Engineering and Emerging Technologies (ICEET), 2021

[11] Vijaya Vardan Reddy S, B. Jaison, A. Balaji, D. Indumathy, S. Vanaja, J. Joselin Jeya Sheela. "Agri-IoT: A Farm Monitoring and Automation System using Internet of Things", 2023 Second International Conference on Electronics and Renewable Systems (ICEARS), 2023

[12] "IoT Based Control Networks and Intelligent Systems", Springer Science and Business Media LLC, 2024