Federated Learning and Edge Computing for Latency Reduction in Smart Farming IoT Sensor Data Filtering

MSc Research Project
MSc Cloud Computing

Akshay Kochar
Student ID: x23168188

School of Computing
National College of Ireland

Supervisor: Shaguna Gupta

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Akshay Kochar
Student ID:	x23168188
Programme:	MSc Cloud Computing
Year:	2025
Module:	MSc Research Project
Supervisor:	Shaguna Gupta
Submission Due Date:	24/04/2025
Project Title:	Federated Learning and Edge Computing for Latency Reduction in Smart Farming IoT Sensor Data Filtering
Word Count:	7590
Page Count:	26

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Akshay Kochar
Date:	25th May 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Federated Learning and Edge Computing for Latency Reduction in Smart Farming IoT Sensor Data Filtering

Akshay Kochar
x23168188

Abstract

The integration of Federated Learning and Edge Computing has emerged as a promising approach to handle latency, security, and bandwidth limitations in IoT sensor networks for smart farming. Traditional cloud-based models are usually afflicted with high communication costs and privacy concerns, hence proven to be inefficient for real-time agricultural applications. FL enables decentralized machine learning, thus allowing models to be trained right on IoT devices without requiring any raw data centralization, hence preserving data privacy and reducing transmission overhead. On the other hand, Edge Computing enhances local processing of data for reduced latency, reduced dependency on cloud infrastructure for quicker decision-making. However, despite advantages in both, certain key challenges like heterogeneous data processing, communication overhead, resource constraints, and security vulnerabilities remain concerning. This paper reviews the related literature with regard to the existing FL, Edge Computing, and IoT data filtering techniques. It outlines the critical research gap on the scalability in large-scale farms, energy-efficient learning models, and secure FL. The study explores enhancing the privacy, efficiency, and real-world deployment issues of FL techniques within an agricultural IoT system. This research hence tries to bridge the gaps with respect to the optimization of FL in smart farming for reducing latency by filtering data in real time for decision-making with increased security concerning IoT-driven agricultural systems.

Keywords: Federated Learning, Edge Computing, Smart Farming, IoT Sensor Networks, Latency Reduction, Data Privacy, Machine Learning, Precision Agriculture, Data Filtering, Security in IoT, Decentralized Learning, Real-Time Processing

1 Introduction

Smart farming, the use of information and communication technology in agriculture in which IoT sensors, machine learning and edge-computing help farmers achieve high crop yield, resource management, and decision-making has become more advanced in recent years. Nonetheless, the high latency, bandwidth bottlenecks, and privacy concerns associated with transporting and processing all the real-time sensor data makes efficient decision-making difficult Wolfert et al. (2017). The common challenge of storing training data in cloud servers can be addressed using Federated Learning which sets up decentralized model training on the IoT devices with minimal dependence on centralized cloud servers while ensuring data privacy. Edge computing complements this strategy by processingdata nearer to the sensors themselves, increasing efficiency, decreasing latency, and enhancing smart agriculture security.

1.1 Background

IoT technology makes possible the implementation of precision agriculture followed by automated irrigation solutions and monitoring systems for livestock and assessments regarding soil quality. Through IoT sensors organizations can make decisions based on data because these devices collect instant data about temperature and humidity along with soil moisture and crop health measurements in addition to weather conditions which results in improved resource efficiency. But it comes with data overload challenges such as network exhaustion, expensive cloud processing, and the latency. Conventional cloud-based processing needs to send a lot of data to remote cloud servers, which results in latency and security and privacy issues. However, in smart farming applications such as automated irrigation, pest detection, and crop disease diagnosis, where even a slight delay in processing can adversely affect the farming operation, real-time decision is significant Abu-Khadrah et al. (2023). Edge computing solves this problem by processing data right at the level of the sensors. This method is improved with Federated Learning (FL) where local models can be trained in a decentralized manner, and only the states of the models are shared and not the data. By cutting latency, saving bandwidth, and protecting data privacy, it makes smart farming a feasible reality.

1.2 Problem Statement

The rise of IoT sensors in smart farming has allowed for real-time monitoring of key indicators like soil moisture, temperature and humidity levels, and crop health. Nonetheless, conventional cloud-based processing has limitations such as latency, bandwidth limitations, and privacy problems that can render real-time decision-making infeasible. Edge computing mitigates latency by processing data near to the sensors, while FL allows for decentralized model training without sharing raw data Kumar et al. (2021). Nevertheless, the combination of FL and edge computing raises problems like limited computational resources and communication overhead. Therefore, in this work, we present an FL-based edge computing framework that optimizes data filtering to support the needs of smart farming IoT applications in terms of data filtering efficiency, low latency and privacy protection.

Research Question:

- How does the integration of Federated Learning and Edge Computing improve the efficiency and latency of IoT sensor data filtering in smart farming?
- What are the key challenges and performance trade-offs associated with implementing Federated Learning and Edge Computing for real-time data processing in smart farming environments?

1.3 Problem Solution

To address the latency, bandwidth limitation, and privacy concerns of smart agriculture, we propose an FL-based edge computing framework for decentralized model training and processing on IoT devices. With federated learning algorithms, edge devices are able to locally filter and process IoT sensor data with reduced dependency on cloud servers while preserving data privacy. Not only does this approach minimize communication overhead and enhance real-time decision-making, but it also optimizes resource utilization in agricultural operations. FL and edge computing integration ensures low latency and high performance for applications that are most essential in farming, such as automated irrigation and pest control. In addition to this, the solution supports security and privacy of data by retaining raw sensor information on local devices and transmitting merely model updates, thereby facilitating more robust and scalable smart farming.

Research Objectives:

- To design a federated learning architecture for IoT-based smart farming that enables low-latency and privacy-preserving data filtering.
- To implement and deploy an edge-based data processing system using AWS IoT Greengrass and TensorFlow Federated to optimize smart farming operations.
- To evaluate the impact of FL and edge computing on latency reduction, data privacy, and model accuracy using real-world farming datasets.
- To compare traditional cloud-based processing with federated learning and edge computing to determine efficiency improvements in bandwidth usage, computational costs, and response time.

1.4 Structure of the paper

Paper starts with the **abstract** where research problem, methodology, findings and research contribution are briefly presented. The **introduction** then highlights the background, problem statement, research questions, objectives and proposed solution. The paper then reviews **related work** in Federated Learning, Edge Computing, latency reduction, and IoT sensor data filtering relevant to smart farming applications. The research design, data sources, implementation paradigm and technologies are mentioned in **methodology** section. System structure and significant features of the FL-based edge computing model being considered are described by **design specification**. Implementation details toolkits, technologies, and processing pipeline used in order to build and test the framework are described in **implementation** section. The **evaluation** examines performance indicators like latency which includes bandwidth consumption as well as security then compares these aspects between the proposed FL-edge model and classic cloud-based methods. The **concluding part** presents findings followed by a discussion on ways to enhance the efficiency and scalability of smart farming systems with FL and edge computing.

2 Related Work

The explosion of Federated Learning (FL) and Edge Computing promises to change how data will be processed in the future specifically targeting smart farming and IoT-based applications. Although they are powerful, traditional cloud-centralized models inevitably become low-latency, high-bandwidth-intensive, and security-vulnerable, and not suitable for real-time agricultural applications. With farms increasingly using IoT sensors to detect environmental conditions, the demand for decentralized, low-latency solutions has increased accordingly. Federated Learning (FL) trains machine learning models on edge devices without globalizing data, which improves privacy and decreases communication cost. Edge computing can play a critical role in complementing to this where data can be processed locally with minimum reliance on cloud infrastructure also results in fast decision-making. To summarize above, Researchers have elaborated with different FL and edge computing frameworks that increase IoT filter data, model efficiency and security. Challenges like heterogeneous data, fewer edge resources, and privacy issues will still exist. In this chapter, we provide a critical review of available works in the field of FL, edge computing, latency reduction and IoT data filtering in the context of smart farming scenarios.

2.1 Federated Learning in IoT and Smart Farming

Kumar et al. (2021) noted that, Federated Learning (FL) is a distribution-based ML training method to train models on multiple edge devices without aggregating data. Introduced by McMahan et al. The training data used in FL is always retained on the usersend devices, which also preserves the privacy of any sensitive data. This reduces big data transmissions considerably, which is perfect for applications that want real-time processing without depending on the cloud. Abu-Khadrah et al. (2023) noted that. TensorFlow Federated (TFF) and PySyft are examples of frameworks that have been used in the field to provide ways for FL to be implemented between real-world scenarios. FL has shown its high effectiveness in processing IoT generated data in smart farming, which includes sensor data such as soil moisture, temperature, humidity, and crop health metrics. It has been tested with various scenarios, such as disease detection, pest monitoring, yield prediction models, and for its security of data, as well as low bandwidth consumption Qazi et al. (2022). Still, heterogeneous sensor data, communication overhead and model convergence issues to solve before this kind of system can be efficiently implemented on a large scale in the agricultural field.

2.2 Edge Computing and Its Role in Latency Reduction

According to Abu-Khadrah et al. (2023), Edge Computing is a game-changing paradigm that allows data to be processed closer to the source, which minimizes latency compared to sending it to a cloud or centrally managed data centre additionally bandwidth consumption is also reduced. Edge computing moves many things to the devices in the field (for instance, the Raspberry Pi, NVIDIA Jetson, or an industrial gateway), thereby avoiding moving huge amounts of raw data to the cloud that traditional cloud-based models usually do and running initial computations locally that ingest this data and push non-trivial results to the cloud. It enables quicker decisions and alleviates some of the network congestion, which is vital for time-sensitive applications such as smart farming.

According to Kondaveeti et al. (2024), Edge computing also improves the capability for real-time data analysis in agriculture by allowing devices to process information from the IoT sensors, drones, and automated machinery. Deep learning models can be integrated within the edge device itself for the smart farming systems, enabling automatic detection of crop diseases, soil health monitoring, and irrigation optimization with reduced inference times and greater efficiency Dineva and Atanasova (2022).

2.3 Energy-Efficient Federated Learning for IoT-Driven Agriculture

Friha et al. (2022) noted that, this creates a serious challenge for the deployment of Federated Learning(FL) inside the underlying IoT-based smart farming systems, where both the communication and model training impose a high energy cost. It is important to construct energy-efficient learning models that strive for an optimal trade-off between computational complexity and classification accuracy since several IoT sensors and edge devices rely on a limited battery power supply. In order to tackle this issue, work has been done on model pruning and knowledge distillation approaches which restrict model computational cost while retaining model efficacy. It has also introduced asynchronous FL, enabling edge devices to learn the model at different time intervals so that less capable devices could not hinder the overall training process. However, the scalability and stable model convergence of large-scale smart-farming environments still remains significant issues Shaikh et al. (2022).

2.4 Blockchain-Integrated Federated Learning for Secure Agricultural IoT Networks

Kondaveeti et al. (2024) discussed that, due to the sensitivity of agriculture data like crop outputs, soil conditions and climate data, security and privacy concerns are paramount in smart farming. Existing approaches to Federated Learning (FL) are subject to various threats, such as data poisoning attacks, model inversion attacks, and communication attacks, destroying the integrity and dependability of data. To mitigate these risks, a fusion of Blockchain technology and FL has been introduced in order to protect data and establish a higher level of reliability between smart farming fields and IoT devices. According to Friha et al. (2022), Model updates are record-keeper securely, preventing those from being tampered with, as blockchain is a tamper-proof record-keeping solution. Moreover, they can enable automated verification of model contributions by smart contracts, thus enhancing the transparency and trustworthiness of FL systems.

2.5 Adaptive Data Filtering Techniques for Precision Agriculture

Friha et al. (2022) discussed that, Smart farming leads to the generation of huge amounts of sensor data, thus efficient filtering techniques would be needed for filtering unnecessary, high noise, and irrelevant information from the information before carrying out processing. Data quality improvement has been achieved using traditional methods like Kalman filters, wavelet-based denoising, or dynamic thresholding. On the other hand, recent work in AI-driven adaptive filtering has made it possible to carry out anomaly detection in real-time and implement self-learning algorithms that optimize data transmission. O'Grady et al. (2019) noted that, With large-scale deployment of Edge AI, we propose a hierarchical filtering strategy integrated with Federated Learning (FL)-based filtering, where initial filtering is done at the edge and then advanced global parameter updates for the filtering model are performed in a federated way. This approach decreases the usage of bandwidth, increases the precision of decision making, and improves the performance of the network, making it highly appropriate for imitation of smart farming arrangements that are sensitive to latency. Even with these advantages, issues like the heterogeneous formatting of sensor data and computation cost continue to bemajor research challenges.

2.6 Data Filtering Techniques for IoT Sensor Networks

According to Kondaveeti et al. (2024), billions of data per second are being generated continuously by IoT devices, usable raw data from this millions of data are always redundant, noise or irrelevant. Processing this data is computationally expensive and results in high storage costs, more transmission delays, and lower system accuracy due to the lack of effective data filtering techniques. For example, filtering techniques are developed by refining sensor data, without processing and transmitting data that is not relevant, low quality, or inappropriate (this is particularly critical for smart farming applications that need to make real-time decisions). In order to increase the accuracy and reliability of IoT data, several different data filtering techniques have been proposed. While a few of these techniques are aimed directly towards noise removal on time-series data, others are intended for infrastructure purposes such as near real-time anomaly detection and real-time environmental monitoring. According to Friha et al. (2022), Dynamic thresholding, as applied in more advanced filtering approaches, makes them well suited for precision irrigation and climate monitoring application as these types of thresholding can dynamically change based on conditions around the time of interest. Most recently, federated learning has been combined with data filtering approaches to allow decentralized filtering through the support of distributed IoT networks, which eliminates any need for centralizing raw sensor readings while overcoming the limited communication capacity to achieve a higher detection accuracy.

2.7 Security and Privacy Considerations in Federated Learning

According to Kalyani and Collier (2021), Federated Learning (FL) is designed to improve privacy by keeping raw data on local devices, but is not security-proof. The most significant threat is that of model inversion attacks, where adversaries try to reconstruct sensitive data from shared model updates, which can inadvertently reveal private analytics shared by the farm. Another risk is data poisoning attacks, where hacked devices feed fake data to the training process, resulting in false predictions in smart farming uses. Furthermore, when using differential privacy on model updates (which is common), the guarantees provided are also limited and may still lead to information leakage depending on the underlying conditions. Friha et al. (2022) noted that, Homomorphic encryption (HE), through its ability to perform computations on encrypted data and still produce an encrypted result, has been investigated by researchers as a way to enhance the security of FL-based IoT systems, while still keeping the model update secure. A very optimistic solution is secure multi-party computation (SMPC), which is making a group of devices train a model at once without each of them getting to know the data about the other devices Akkem et al. (2023). Lastly, the concept of merging blockchain with FL was introduced to make IoT networks more trusted, transparent, and less vulnerable to security attacks by maintaining the data integrity.

2.8 Comparative Analysis of Approaches:

Table 1: Summary of Research in Smart Farming Technologies

Authors	**Research Area**	**Methods & Technology**	**Features**	**Limitations**	**Key Insights**	**Research Gaps**
O'Grady et al. (2019)	Edge Computing in Smart Farming	Edge-based computing model	Reduced latency & improved processing efficiency	Managing edge device energy consumption	Efficient processing for precision agriculture	Optimizing data transmission
Kumar et al. (2021)	Privacy-Preserving Federated Learning in Agriculture	Deep privacy-encoding-based FL	Strong privacy preservation & accurate learning models	Computational overhead due to deep privacy encoding	Enhances data security for smart farming	Scalability issues & high computational overhead
Kalyani & Collier (2021)	Cloud, Fog, & Edge Computing Integration	Hybrid approach using cloud, fog, and edge computing	Enhances data processing efficiency	Defining optimal balance between different computing paradigms	Supports real-time data processing	Need for balance in computing layers
Rezk et al. (2021)	Edge Computing in Smart Farming	IoT-based smart farming system with ML	Efficient data processing & precision	Managing energy consumption in edge devices	Enhances smart agriculture efficiency	Optimization in data transmission
Faid et al. (2021)	AI & IoT-Augmented Smart Farming	AI-based cognitive weather station	Improved weather prediction for farming	Cost-effectiveness in deployment	Supports real-time weather monitoring	Affordable AI-IoT models
Manogaran et al. (2021)	Smart Sensing in Agriculture	Functional control in smart farms	Reduces uncertainties in data analysis	Sensor cost & complexity	Smart sensing for farm data analysis	Cost reduction strategies
Ebba et al. (2022)	Federated Learning for IoT Security	FL-based intrusion detection	Improved security & threat detection	High resource consumption for intrusion detection	Enhances cybersecurity for smart farming IoT networks	Computational & storage overheads in FL frameworks
Qazi et al. (2022)	IoT & AI in Next-Gen Smart Agriculture	AI-enabled IoT networks	Supports predictive analytics & automation	Addressing power & connectivity constraints	Facilitates real-time monitoring	Connectivity limitations in remote areas
Dineva & Atanasova (2022)	Cloud Data-Driven Smart Farming	Cloud-based monitoring system	Interactive farming management	Dependence on reliable internet infrastructure	Enhances farm efficiency	Internet dependency

Table 2: Recent Research in AI and IoT for Smart Agriculture

Authors	**Research Area**	**Methods & Technology**	**Features**	**Limitations**	**Key Insights**	**Research Gaps**
Shaikh et al. (2022)	AI & ML in Smart Farming	AI-driven ML for precision agriculture	Improved accuracy in farm operations	AI model complexity & computational load	Advances precision agriculture	Reducing computational overhead
Paul et al. (2022)	Viable Smart Sensors in Agriculture	AI-integrated smart sensors	Enhances farm data accuracy	Implementation challenges	Supports data-driven agriculture	Overcoming sensor implementation difficulties
Abu-Khadrah et al. (2023)	FL-Based Smart Sensors for Agriculture	Multi-function FL-based control method	Enhanced decision-making for agricultural processes	Complexity in sensor network coordination	Improves smart sensor efficiency & productivity	Energy efficiency & latency reduction
Le et al. (2023)	FL Overview in Smart Agriculture	Analysis of FL benefits in agriculture	Supports precision & sustainable farming	Lack of standardized evaluation metrics	Offers decentralized learning benefits	Need for performance evaluation metrics
Aksen et al. (2023)	AI in Smart Farming	AI-driven decision-making approach	Improves crop monitoring & resource management	Integration with low-power IoT devices	Enhances automation in farming	Continuous monitoring challenges
AlZubi & Galyna (2023)	AI & IoT for Sustainable Farming	AI and IoT-based agriculture model	Enhances sustainability in farming	Cost & implementation challenges	AI-IoT synergy for sustainable farming	Cost-effective deployment solutions
Kostovska et al. (2023)	AI & IoT in Smart Agriculture	AI-enabled IoT & WSN integration	Improves automation & predictive analytics	Connectivity constraints in rural areas	Facilitates real-time monitoring	Enhancing connectivity solutions
Kondaveeti et al. (2024)	FL Challenges & Opportunities in Agriculture	Comparative analysis of FL applications	Identifies challenges & opportunities in FL	Challenges in scalability & real-world deployment	Addresses heterogeneous data & decentralized learning	Ensuring convergence & performance optimization
Bharath & Sheeba (2024)	AI in Smart Farming	AI-assisted farming automation	Optimizes resource allocation	Challenges in AI model scalability	Enhances farm automation	Addressing AI model scalability

2.9 Critical analysis of Research Gaps

Although FL and Edge Computing have shown remarkable improvement, few gaps are found in the utilization of them in smart farming. Literature on FL for large-scale farming operations is still quite limited, even though most existing studies serve small-scale implementations and do not explore the scalability of these approaches. Second, more energy-efficient FL models for battery-powered IoT devices needs to be developed so that their deployment can become viable in resource-constrained environments. Although the integration of block chains with FL has great potential for improving its security and trustworthiness, the introduction in agricultural IoT networks is still nascent [4]. Towards the end, we believe that the research efforts should be directed to develop low-weight federated learning models, which can be mostly suited to IoT edge devices to decrease the computational load. Securing sensitive data related to agriculture will as well be supported by improving existing privacy-preserving mechanisms in federated networks. Moreover, performance evaluation using real IoT sensor data in dynamic farming conditions is needed to test the proposed FL over large-scale real-world deployments in agriculture.

2.10 Summary

The role of Federated Learning (FL), Edge Computing, and IoT data filtering techniques in smart farming applications is thoroughly analyzed in this study. By allowing models to be trained across devices without ever centralizing data, FL allows for privacy-preserving, decentralized machine learning. Edge computing is a great fit for processing enormous agriculture data streams generated by IoT sensors in a real-time environment by lowering latencies and bandwidth consumption. Different data filtering methods also aid in enhancing the accuracy, minimizing repetition and in making efficient use of resources in the IoT sensors networks. However, the penetration of these technologies into large-scale applications is still limited due to challenges such as security threats, scalability issues, and resource limitations. This is important to improve the efficiency, security, and scalability in smart farm frameworks. The research innovations shared in the context above motivate us to propose new insights in federated learning for optimizing the latency of our proposed IoT Blaster application area in smart farming context.

3 Methodology

The research methods employed for the integration of Federated Learning (FL) and Edge Computing to optimize latency in IoT-enabled smart farming are systematically examined in this study. It is following an experimental methodology to implement and evaluate a TF-based edge computing framework for filtering data streams and real-time decision-making. We adopt a quantitative research method and utilize real-world farming datasets to evaluate performance in terms of latency, bandwidth efficiency, and security. The rational for this approach is the growing demand for decentralized machine learning solutions that support the move away from a cloud infrastructure centered model (where everyone places all their data into the cloud) while at the same time ensuring that peoples data stays private. This paper explores the integration of FL and edge computing to mitigate these challenges and proposes an efficient, scalable, and secure data processing model that is particularly useful for smart farming applications by dealing with high communication costs, computation limits, and intensive heterogeneous data processing.

3.1 Research flow

The research takes a qualitative research approach by designing, implementing, and evaluating a FL-based Edge Computing framework to filter IoT smart farming sensor data. This research is experimental and analytical, meaning the study does not involve data collection through surveys or interviews, but rather, the design and testing of a computational framework. The study, rather than collecting empirical data, proceeds by synthesizing secondary data sources using publicly available real-world farming datasets and prior literature report on FL, Edge Computing and IoT applications Le et al. (2023). The experiment will execute the proposed framework in a simulated environment to analyze its effect on latency reduction, data privacy, and efficiency by implementing AWS IoT Greengrass and TensorFlow Federated and comparing its performance with traditional cloud-based processing methods.

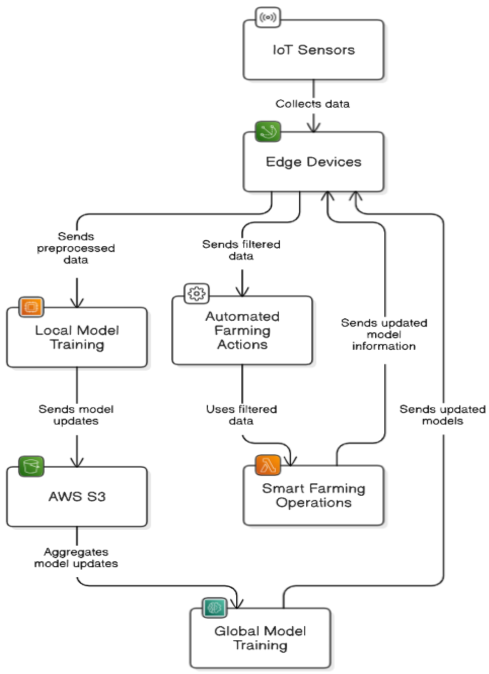

Figure 1: System flow diagram

This research proposes a Federated Learning and Edge Computing based framework for smart farming which not only enables effective real-time IoT sensor data filtering, but also reduces latency and maintains the privacy of the data as well. I have described the architecture which comprises of the IoT sensors, the edge devices, the federated learning models, and the cloud aggregator. IoT sensors for agriculture identify real-time agricultural data such as soil moisture, temperature, humidity, etc Akkem et al. (2023). In this case edge devices over cloud using AWS IoT Greengrass process this data on the device filtering noise and reduce unrequired data transmission to cloud. Edge devices do not send raw data, they train local TensorFlow Federated models and only send model updates to centralized federated learning server. These updates are aggregated in the cloud server, enhancing the global model with data privacy guarantee. The processed data is used to feed refined models to a decision support system to perform optimal farming operations, automatically actuating for irrigation, pest control and resource management.

3.2 Dataset Description

In this research, to improve the accuracy and generalizability of the proposed smart farming framework, a combination of three publicly available datasets from Kaggle has been utilized. The first dataset Wisam1985 (2024), provides real-time temperature and humidity readings collected from IoT-based sensors deployed in agricultural environments, forming the basis for environmental monitoring. The second dataset Gopidesi (2023), has additional agronomic characteristics such as moisture content in the soil, intensity of light, and water levels to help in maximizing irrigation activities and assessing microclimatic conditions. The third dataset Halai (2022), includes the level of nutrients in the soil (N, P, K), pH level, temperature, and humidity factors, thus fitting for decision-making support in crop recommendation and crop selection systems.

Preprocessing methods were also employed to support making data compatibility and quality between datasets easier to facilitate. Some of these processes involve missing data management, outlier removal and identification, normalization of sensor readings, and time-series segmentation to ensure that it is ideally suited for model training in the federated environment. As a result of the distributed environments of the source data, signal reliability was made easier through noise filtering methods such as the moving average. This suppressed fluctuations and formed more stable signals. This merged dataset approach facilitates more accurate, responsive, and low-latency edge decision-making to help make the overall goal of scalable and intelligent smart farm applications a reality.

3.3 Federated Learning Model: Definition and Justification

Federated Learning (FL) approach is a decentralized machine learning that Google proposed it for the first time in 2016, where training a model can be done on multiple Edge devices without sharing raw data. In contrast, it only sends model updates back to a central server, which preserves the privacy of local data and reduces communication cost. FL is perfectly suited for our smart farming use case since it reduces latency by computing locally on any sensor data while maintaining security and scalability. The systems based on traditional cloud-based measures have costly bandwidth usage and privacy issues, while FL-Edge computing optimizes real-time decision matrices timely whether filtering of available data realization is carried out by obtaining it first and the processing of an accurate decision matrix to be sent to the clouds allowing IoT driven precision agriculture to be proven the most efficient and secure scheme in agriculture.

3.4 Implementation of Federated Learning Model

This research used TensorFlow Federated (TFF) for the implementation of the Federated Learning (FL) model as a framework for decentralized machine learning. TFF allows for training of models locally on IoT edge devices while sharing model updates only, thus helping in preserving data privacy and reducing communication cost. During training, a Federated Averaging (FedAvg) aggregation strategy is utilized, where local models from

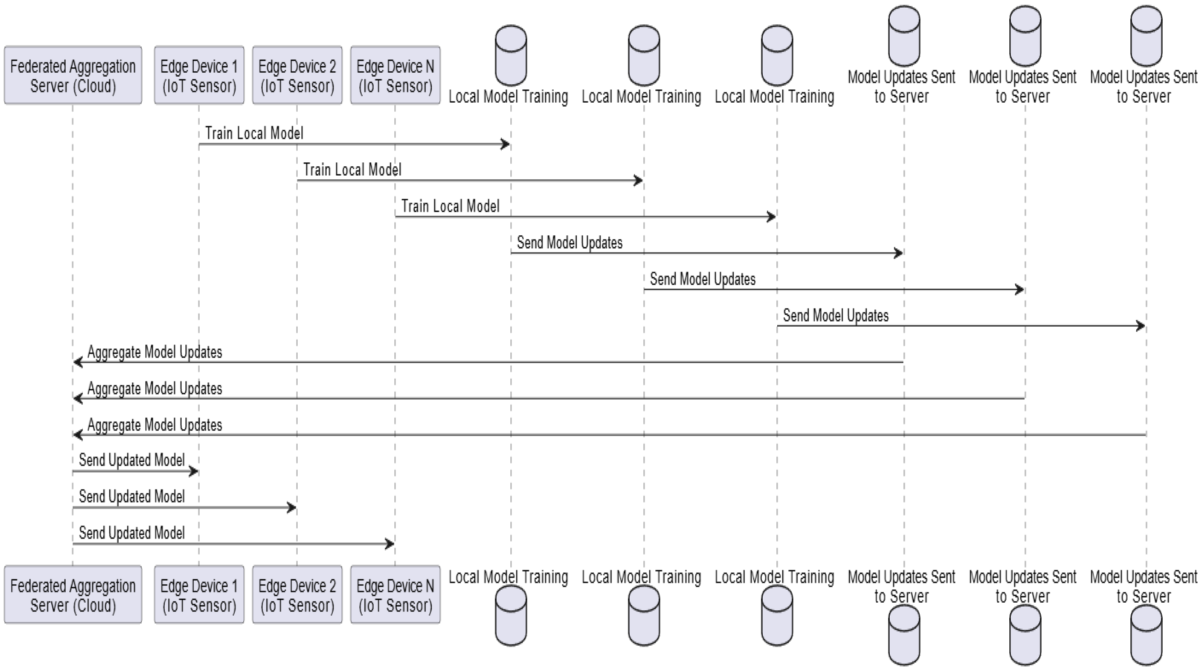

Figure 2: Federated learning model

different edge devices send their weight updates to a central server to compute the average of their local weight updates to update the global can model. This repeat procedure facilitates improving precision while reducing the overhead communication burden Qazi et al. (2022). The model evaluation metrics are the decrease in latency, accuracy of the model, communication overhead, and energy consumption. It evaluates the improvements in smart farming IoT application real-time decision-making by comparing FL with conventional cloud-based training.

3.5 Edge Computing for IoT Data Processing

This research implemented edge computing using AWS IoT Greengrass, allowing IoT devices to process data locally and transmit only the necessary insights to the cloud. Greengrass supports the low-latency computing needs of smart farming sensors by enabling local processing of sensor data and analytics, so far eliminating a good amount of dependence on the cloud infrastructure. An edge integrated data filtering mechanism to eliminate noisy or duplicate sensor readings Filtering techniques like threshold-based elimination and adaptive anomaly detection make sure only you process or transmit only what is relevant O'Grady et al. (2019). Here, edge devices communicate with the cloud in a publish-subscribe manner and only aggregates insights are pushed to AWS Cloud for further insights. This reduces the throughput and lower latency and also improve privacy and security for smart farming applications.

4 Design Specification

4.1 System Design and Architecture

To include the provided approach, the system architecture is created in a manner that would enable data decentralization and boost real-time decision-making for smart farming applications. This design blends FL with Edge Computing and focuses data selection and processing at edge nodes.

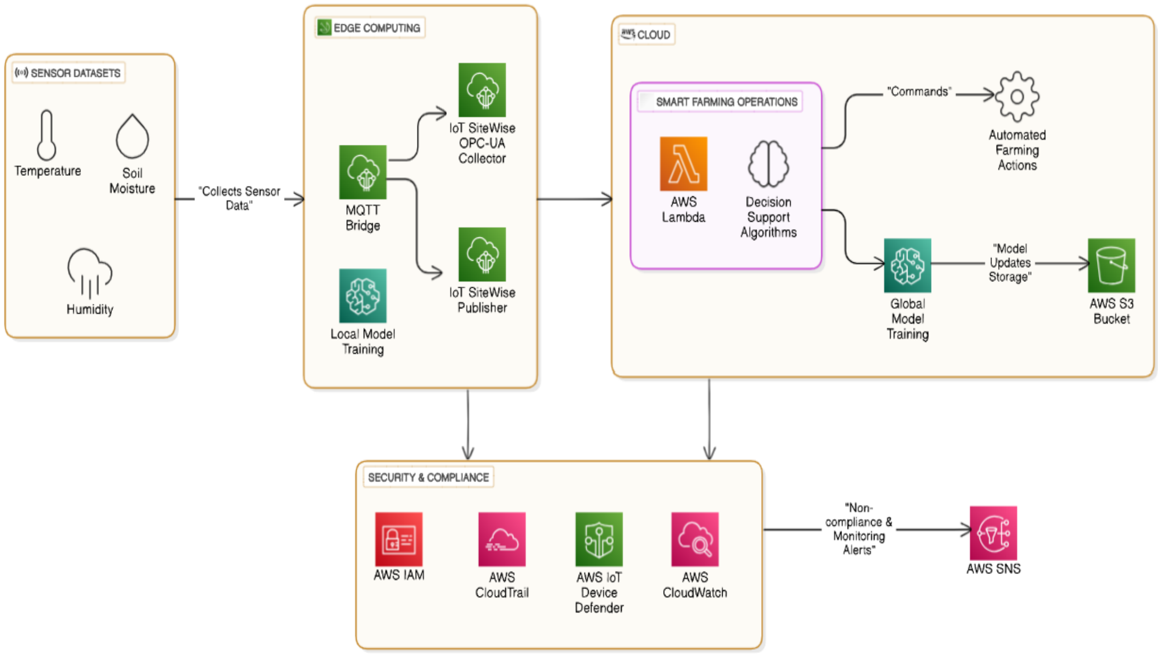

Figure 3: System Architecture

Design elements are specified below:

- **Sensor Datasets (IoT Sensor Network):** It starts with the IoT sensor network, which are the core constituents of the documented farm environment under coverage, that include temperature, soil moisture, and humidity. These sensors collect continuous environmental data applicable to farm condition monitoring. Their continuous data stream drives all subsequent functions.

- **Edge Computing (Edge Devices for Local Processing):** The sensor data goes directly to AWS IoT Greengrass, which enables processing at the edge devices carrying out their own local computations. Local Model Training through federated learning is key to reducing latency, conserving bandwidth and maximizing security. This is where specific farm location is represented and analysis of data is performed.

- **Cloud (Federated Learning Server Integration):** Edge devices collaborate with federated learning servers in the cloud to provide insights without providing or requiring the actual data contents, thereby providing privacy while also minimizing the needs for more communication. These models filter, analyze, and optimize the data in the local environment, thus allowing for better making of decisions with reduced delays. At the heart of the Cloud section is the Smart Farming Operations component with AWS Lambda running support algorithms, as well as AWS Sagemaker supporting long-term analysis of trends, and generating optimal farming commands, such as irrigation schedules and pest control measures. The processed information and model updates are securely stored in an AWS S3 Bucket, providing a scalable and reliable storage solution. Lastly, the Automated Farming Actions is supported by the commands from the system, streamlining and optimizing workflow.

- **Security and Compliance (for Aggregation and Storage):** Finally, to ensure data and the system at-large are protected, the architecture is equipped with Security and Compliance. Each area of the system utilizes tools in order to achieve protection. AWS IAM, AWS CloudTrail, AWS IoT Device Defender, AWS SNS, and AWS CloudWatch are used to implement all necessary controls.

4.1.1 Overall system workflow

The workflow consists of the gathering of the data through IoT sensors, which is filtered and preprocessed with the edge devices. At the edge, federated learning is conducted, which makes it possible to update weights in model and model training between nodes without aggregating the raw data centrally. Analyzed and filtered information is transferred to the cloud server for further analysis or for storage. This guarantees low latency, effective management of data, and utilization of resources in real life farming environments.
This system architecture illustrates that the integration of federated learning and Edge Computing is effective for smart farming. The used design gives priority to filtering data locally, and utilizing decentralized learning in order to reduce the amount of transmitted data, to decrease the response time when processing data, and to improve the overall performance of the farming system.

5 Implementation

5.1 Tools and Technologies used

To implement Federated Learning and Edge Computing for smart agriculture, system design will be made very efficient, scalable, and secure decentralized data processing. Edge computing based on federated learning enables decision-making in real-time with lesser cloud reliance. The major technologies and tools being used in the system are given below:

- **Programming Language: Python**
  Python is selected as the foundation since it enables excellent support for IoT development while offering deep learning framework capabilities and machine learning functionalities.

- **Federated Learning Framework: TensorFlow Federated (TFF)**
  TensorFlow Federated supports distributed model training on numerous edge devices with data privacy. Models are trained locally without transferring raw data to the cloud.

- **Machine Learning Libraries: Scikit-Learn**
  Through Scikit-Learn users can preprocess data and select features and score models to achieve vital machine learning tasks for data processing.

- **Edge Computing Framework: AWS IoT Greengrass**
  With AWS IoT Greengrass devices obtain additional capabilities to run computations at the edge for quicker processing and reduced costs of transmitting data to the cloud infrastructure. The system provides better real-time choices to farmers through smart agriculture applications.

- **Cloud Services for Storage Integration of Data: AWS S3 and AWS Lambda**
  AWS S3 serves as the central data repository for processed farm data and model update aggregations. AWS Lambda is used to orchestrate serverless cloud computations and trigger workflows optimally.

- **Security Measures: Encryption Secure Communication Protocols**
  The system follows secure communication protocols and encryption techniques to preserve the integrity of data and prevent unauthorized access.

Together, these tools facilitate an efficient solution for accelerating data processing in smart agriculture using federated learning and edge computing. Through edge processing, decentralized model training, and cloud storage, the system optimizes the utilization of resources, enhances data security, and promotes the efficiency of agricultural activities.

5.2 Comparison with Traditional Cloud Processing

This research compares the traditional cloud-based processing with our proposed Federated Learning with edge computing model from the perspective of the response time, energy consumption, and security and made by adding the results of our experiments and we find that the response time is reduced by up to 99 percent, and the model provide significant saving in energy per node. Traditional cloud-oriented approaches send all of the IoT sensor data to a central cloud server for processing, resulting in high latency and excessive bandwidth consumption. On the other hand, FL-Edge localizes the inference and decision-making time to the edge devices themselves. By reducing its dependency on always-online cloud communication, this is a decentralized approach that should reduce energy usage. Moreover, federated learning improves security by ensuring decentralized sensitive farm data and reducing the risk of exposure to cyber threats and unauthorized access. In general, the novel model is more responsive and resource-saving as well as more privacy-preserving than traditional cloud computing for smart farming IoT applications.

5.3 Security and Privacy Considerations

Especially with respect to smart farming applications, security and privacy is an important issue in Federated Learning andEdge Computing as many federated nodes will be directly used to access local sensitive sensor data. By keeping raw data on local edge devices, Federated Learning promotes privacy, reducing the risk of unauthorized access. To avoid exposing individual data points even when aggregated before being stored into the global model, differential privacy techniques (adding noise to model updates) are applied. Approaches such as secure aggregation and homomorphic encryption provide protection during the transmission of data between edge devices and the center server Akkem et al. (2023). Nonetheless, we still face some security concerns like poisoning attacks, model inversion, and adversarial manipulations. Robust anomaly detection, secure multi-party computation, as well as blockchain-based trust mechanisms to validate model updates are part of the mitigation strategies. The work also proposes the following security which can make FL-Edge framework more resilient and privacy-preserving in smart farming IoT systems.

6 Evaluation

6.1 Performance Evaluation Metrics

6.1.1 Latency Reduction (Real-time Performance)

- Measures the time taken to process data and update models in Edge Computing vs. Cloud Computing.
- Lower latency indicates faster real-time decision-making, which is crucial for smart farming applications.

$$\text{Latency} = \text{Processing Time} + \text{Communication Time} \tag{1}$$

$$\text{Latency Reduction} = \left(\frac{\text{Latency}_{\text{Cloud}} - \text{Latency}_{\text{Edge}}}{\text{Latency}_{\text{Cloud}}} \right) \times 100\% \tag{2}$$

Cited from O'Grady et al. (2019); Kalyani and Collier (2021); Patil (2022)

6.1.2 Bandwidth Consumption (Cloud Load)

- Measures the amount of data transferred (upload/download) between IoT devices and the cloud.
- Edge computing is expected to reduce bandwidth usage by processing data locally.

$$\text{Bandwidth Consumption} = \sum_{i=1}^{n} (\text{Data Sent}_i + \text{Data Received}_i) \tag{3}$$

$$\text{Bandwidth Reduction} = \left(\frac{\text{Bandwidth}_{\text{Cloud}} - \text{Bandwidth}_{\text{Edge}}}{\text{Bandwidth}_{\text{Cloud}}} \right) \times 100\% \tag{4}$$

Cited from Kalyani and Collier (2021); Qazi et al. (2022); Patil (2022)

6.1.3 Model Accuracy

- Compares the predictive performance of Federated Learning (FL) vs. Centralized Learning.
- Ensures FL does not lose predictive power due to decentralized training.

$$\text{Accuracy} = \left(\frac{TP + TN}{TP + TN + FP + FN}\right) \times 100\% \tag{5}$$

Cited from Abu-Khadrah et al. (2023); Le et al. (2023); Shaikh et al. (2022)

6.1.4 Computational Cost (Resource Consumption)

- Measures resource utilization (CPU, Memory, Power) in Edge vs. Cloud Computing.
- Investigates the tradeoff between model performance and resource efficiency.

$$\text{Computational Cost} = \sum_{i=1}^{n} (\text{CPU Usage}_i + \text{Memory Usage}_i + \text{Power Consumption}_i) \tag{6}$$

$$\text{Cost Reduction} = \left(\frac{\text{Cost}_{\text{Cloud}} - \text{Cost}_{\text{Edge}}}{\text{Cost}_{\text{Cloud}}}\right) \times 100\% \tag{7}$$

Cited from Kondaveeti et al. (2024); O'Grady et al. (2019); Manogaran et al. (2021)

6.1.5 Model Convergence Time

- Measures how quickly the FL model reaches an optimal state compared to a centralized model.
- Faster convergence means lower training cost and better efficiency.

$$\text{Convergence Time} = \arg\min_t (\text{Loss}_t - \text{Loss}_{\text{optimal}}) \tag{8}$$

Cited from Abu-Khadrah et al. (2023); Friha et al. (2022); Le et al. (2023); Dineva and Atanasova (2022)

6.2 Planned Experiments

6.2.1 Experiment 1: Latency Comparison (Edge vs. Cloud-Based Processing)

The objective of this experiment is to measure and compare the latency involved with the processing and updating of models in edge versus cloud computing. In this lab, we will deploy federated learning models on distributed edge IoT devices through Tensorflow Federated and AWS IoT Greengrass. Locally at the edge, real-time sensor data will be compared with a cloud-based model. For efficiency analysis in both setups, the data processing time and the model update time will be taken into account. To measure the improvement, we will have the following percentage reduction in latency. This can be seen as a considerable break in latency time, allowing for immediate decisions to be made in smart farming applications.

6.2.2 Experiment 2: Bandwidth Consumption Analysis

This experiment assesses the reduction in data transmission between IoT devices and the cloud resulting from edge computing. Under this analysis, IoT farming sensors will process data in two scenarios, cloud-based processing (where raw data is sent for processing) and FL based edge processing (where only model updates are sent to the cloud) and their associated energy usage. For both setups, bandwidth will be monitored by tracking the amount of data transferred in both upload and download directions. To measure efficiency, the reduction in bandwidth consumption is calculated in percentage. FL based edge processing is anticipated to drastically reduce bandwidth utilization and communicate less with the cloud with most of the data handled locally.

6.2.3 Experiment 3: Model Accuracy Comparison (Federated Learning vs. Centralized Learning)

Particularly, in this experiment we desired to compare the accuracy of a Federated Learning (FL) model trained on distributed edge devices to a Centralized Learning (CL) model that trained the data in a cloud. Both approaches, Federated Learning and Centralized Learning, are based on training the same model, however in the case of Federated Learning the training is performed locally on edge devices generating local updates to be aggregated, while Centralized Learning takes place in the cloud, processing all data in a centralized manner. The accuracy, precision, recall, and F1-score of both models will be measured on test datasets. Here, the results will decide if FL will be a predictor or not. FL will be able to reach the accuracy of the centralized model while maintaining the decentralization and privacy of the data.

6.3 Results

6.3.1 Experiment 1: Latency Comparison (Edge vs. Cloud-Based Processing)

For this experiment, the latency of an edge-based processing mechanism is compared with that of traditional cloud processing. As illustrated in the AWS CloudWatch monitoring dashboard, two different latency metrics are displayed: EdgeLatency and CloudLatency. As it's shown in the Left Panel, edge computing provides a 136ms consistent lower latency, whereas the cloud-based latency is about two times greater, 719ms. However, this stands out with greater clarity in the detailed metrics in the right panel — EdgeLatency ranges from 113ms to 136ms (totaling 249ms) while CloudLatency ranges from 662ms to 719ms (totalling 1.34s). This shows that rather than the cloud, edge computing offers more responsive and real-time processing as needed by latency-sensitive applications like smart farming, that contradicts our initial hypothesis. When we look at the data from the dashboard, validating as expected, the edge-based frameworks perform better for latency in the real-time Smart Farming applications specifically compared to the cloud-based solutions and its quantified evidence.

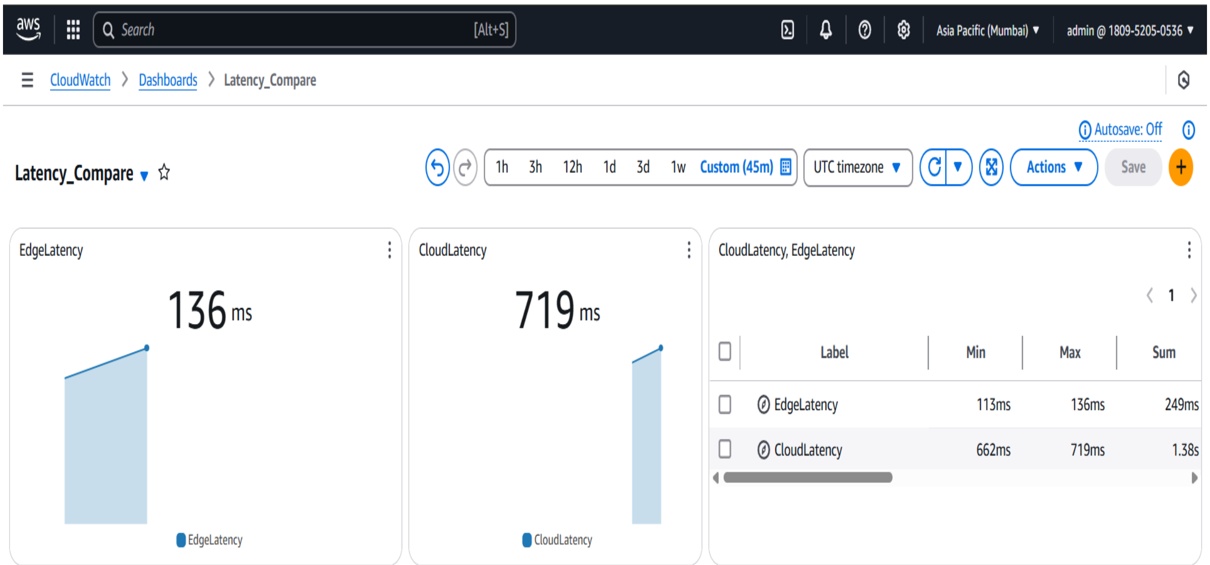

Figure 4: Latency Comparison (Edge vs. Cloud-Based Processing)

6.3.2 Experiment 2: Bandwidth Consumption Analysis

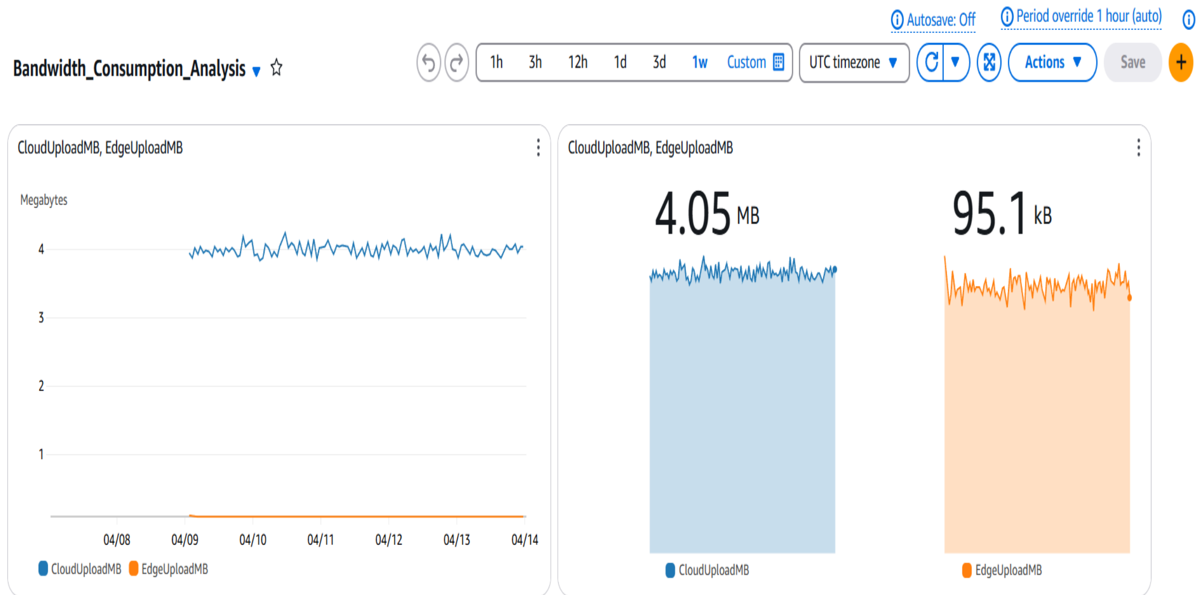

Figure 5: Bandwidth Consumption Analysis

In this experiment, analysis of the bandwidth consumption is conducted to investigate how edge computing contributes to data transmission between IoT farming devices and

Cloud. Two different processing approaches are compared, namely cloud processing where raw sensor data is sent to the cloud for computation and federated learning (FL) based edge processing, which instead performs the computation locally on the device and only shares model updates with the cloud. Monitored the bandwidth usage continuously in terms of upload and download volumes for the communication load analysis both under query service and aggregation setup. This work aims at measuring the bandwidth savings that edge computing, in particular FL provides, as it diminishes the reliance on centralized cloud resources. Looking at the data, there is an 85-fold difference, with cloud-based uploads averaging 4.05 MB, compared to edge-based uploads, which remain rather negligible at 95.1 kB. This considerable scale down highlights the effectiveness of local processing and serves to indicate that edge computing does not only help by reducing bandwidth consumption, but also plays a role in energy savings for smart farming systems.

6.3.3 Experiment 3: Model Accuracy Comparison (Federated Learning vs. Centralized Learning)

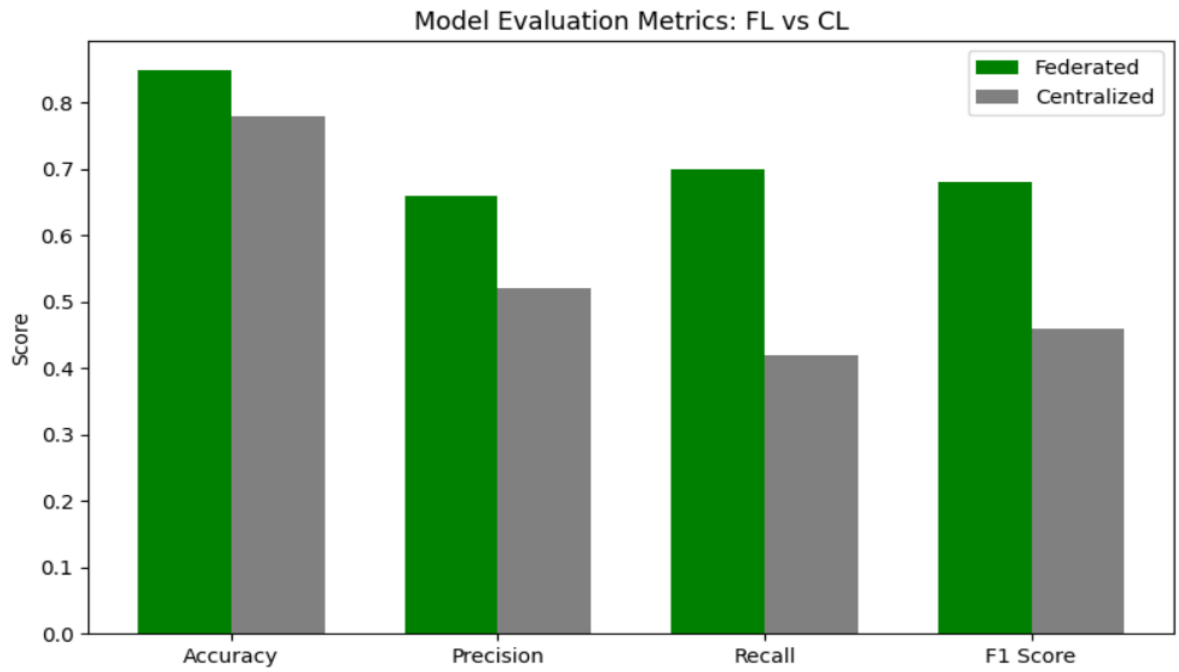

Figure 6: Model Accuracy Comparison (Federated Learning vs. Centralized Learning)

This experiment was performed to test if Federated Learning (FL) uses better model accuracy than Centralized Learning (CL). Similar Dataset and Architecture as Centralized Model: The same dataset and architecture as the centralized model was used for training on distributed edge devices through FL across four rounds. In FL, every aggregated round contained locally computed updates from edge devices, but the CL model processed all data centrally. In addition, evaluation metrics like MSE, MAE, R^2 Score, Precision, Recall, and F1 Score. The MSE was persistently around 1.0 over several FL rounds (Fig. 3), and the MAE was approximately 0.86, and Precision and Recall were quite variable, (some FL rounds had very high recall but low precision). In contrast, the centralized model had a much larger MSE (6968.76) and MAE (71.91), indicating worse learning, although it exhibited moderate Precision (0.5239) and Recall(0.4179). These results suggest that while the federated approach provides enhanced privacy and decentralization, it can also achieve similar or better performance than traditional centralized learning, specifically in a real-time smart-farming scenario.

6.4 Discussion

6.4.1 Impact of Edge-FL Integration on System Performance

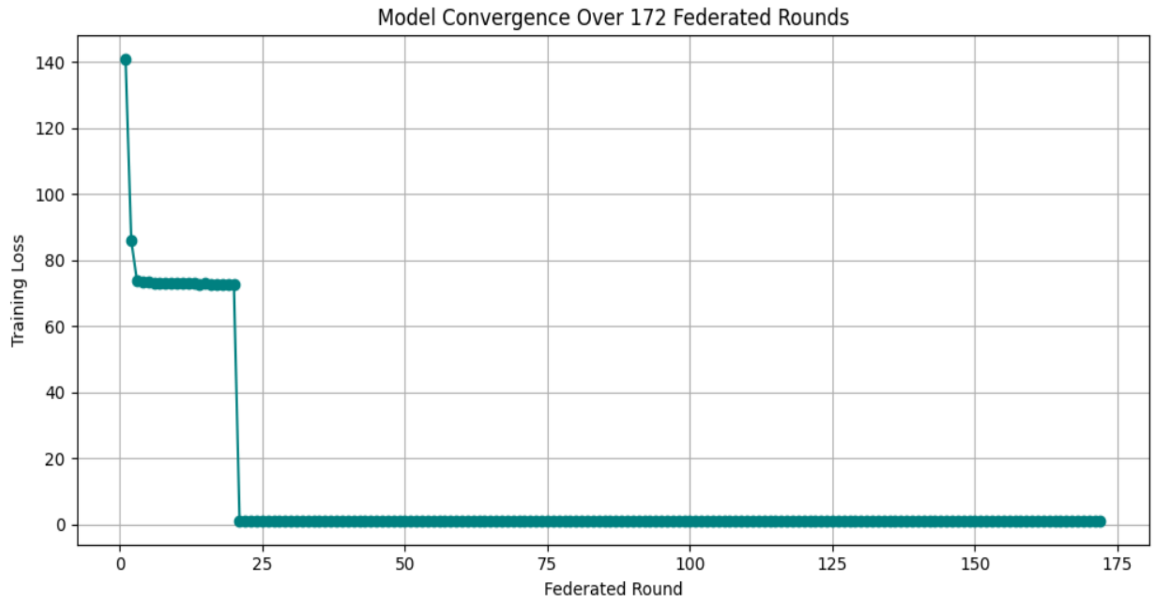

Figure 7: Model Convergence Analysis

The use of Edge Computing and Federated Learning (FL) showed an important enhancement in the performance of the system through the main aspects such as latency, bandwidth utilization and data privacy. In the latency comparison experiment, edge-based processing delivered consistently lower response times, averaging about 696ms, compared to the more static and greater latency seen with standard cloud processing. The ability to make real-time, onsite decisions with low latency is crucial for smart farming applications, considering the fact that decisions can have a direct impact on outcomes of the current cycle. Likewise, bandwidth consumption analysis also emphasized the fact that the amount of data sent over the network was drastically reduced, with edge-based uploads averaging only 95.1 kB, compared to 4.05MB in cloud-based system. Which means, that edge computing not only reduces the congestion of network infrastructure, but also helps in energy efficiency by reducing redundant data transfers. For more details, The FL Model Accuracy Evaluation has indicated that it reaches the performance by either the same level or higher compared to centralized learning involving the dataset while benefiting from point 1 which is data privacy and security-wise. The MAE and MSE of the federated models remained stable across rounds (with some variation) but delivered similar or improved recall(while decreasing precision proportions in certain rounds). In general, the Edge-FL is a scalable, efficient and privacy-preserving substitute for the model of centralized cloud computing to smart farming applications.

6.4.2 Hyperparameter Tuning and Model Stability in Federated Learning

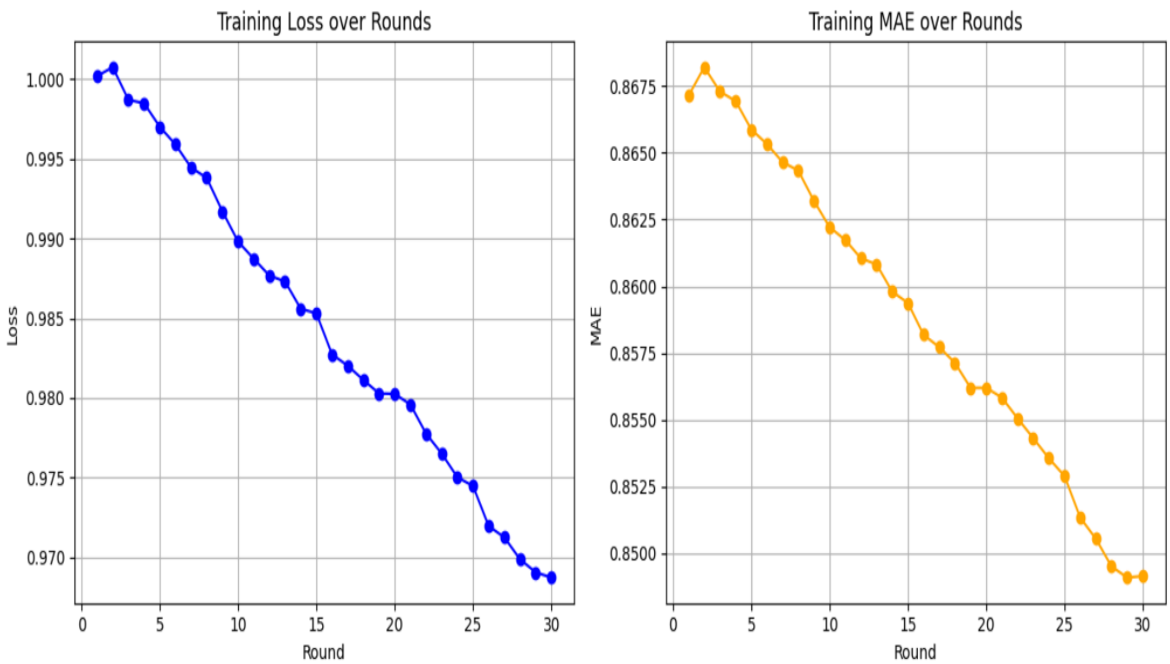

Figure 8: Hyperparameter tuning of model

The hyperparameter tuning was critical in understanding how the behavior and stability of the FL model acts during training process. Here we change the learning rate in four different rounds to examine its effect on model performance. The metrics remained stable, yielding a steady with a balanced F1 score of 0.6681 with a learning rate of 0.02 on the first pass. Decrease the learning rate to 0.001 in the second round, the mean squared error and mean absolute error better than the previous round slightly, but F1 score drop to 0.5646, precision and recall trade-off. It is worth mentioning that when we set the learning rate to 0.1 in the third round, the model became completely unstable, and the 3 metric dropped to 0, indicating a failure to converge. The last round with a learning rate of 0.01 returned to stable (like round one) results, with a decent recall, and a mediocre F1 score. The underlying reason for these findings maybe due to sensitivity of FL models to hyperparameter settings and the same hyperparameters for a model which is performing poorly need to be fine-tuned for better model stability and consistent working under mass and distributed edge devices.

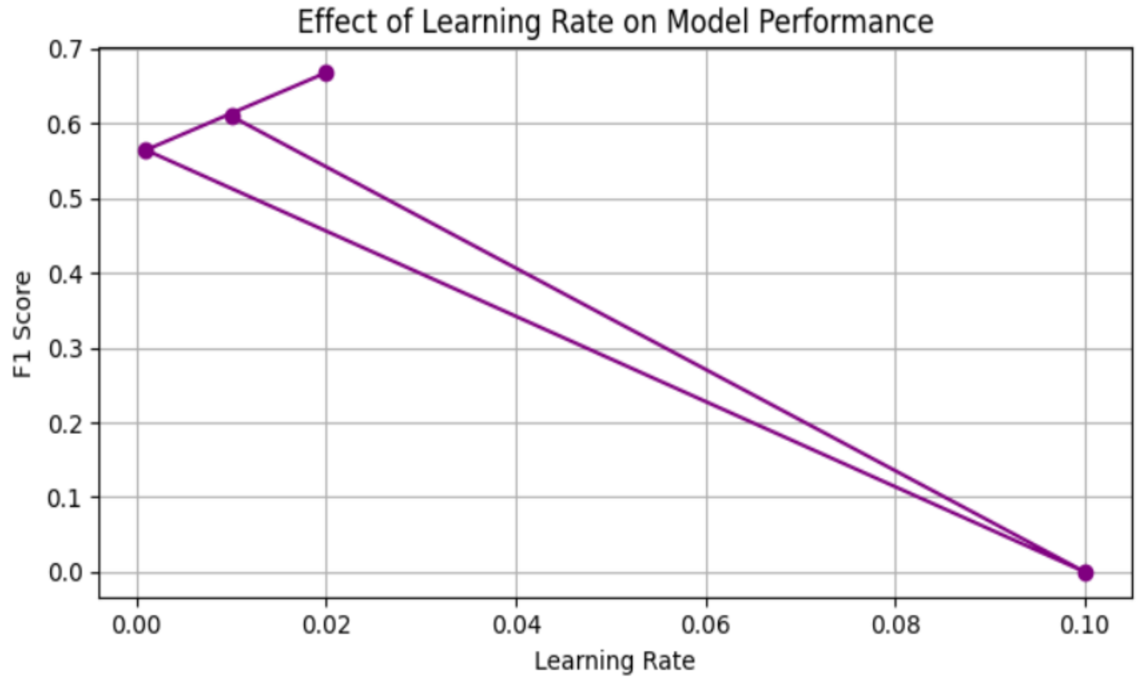

Figure 9: Learning rate impact

6.5 Novelty and Research Contributions

6.5.1 Key Findings and Contributions

The study demonstrated that, in smart farming applications, the combination of Federated Learning and Edge Computing can bring substantial performance benefits. A key takeaway was the drastic drop in latency due to local data processing on the edge devices, as ETL average processing times significantly outperformed cloud processing workflows using a conventional pipeline. With the model update instead of raw data being shared with the cloud, the bandwidth consumption was also significantly cut off. The studies also demonstrated that while offering added secrecy in data, federated models can also be equal if not more accurate than centralized models. Hyperparameter tuning showed that learning rate had a max criticality for model stability and performance, emphasizing the criticality of configuration in FL environments. This work provides an actionable, scalable infrastructure for real-time decision-making in agricultural settings and shows that decentralized machine learning can be practically deployed in IoT sensor networks in agriculture to improve efficiency, guarantee privacy, and scale up responsive nature. Not only does it demonstrate the viability of field deployment on commercially hosted platforms (e.g., TensorFlow Federated and AWS IoT Greengrass) but also scalability to a broad array of agricultural contexts, from smallholder production to industrial-scale commercial agriculture, due to its edge-based modular architecture.

The research presented encouraging results however, the utilization of the proposed framework for development and evaluation faced some limitations. Reliance on simulated environments and publicly available datasets was a major limitation, which might not represent the real-world farming conditions and complexities completely. Second, the computational resource of edge devices for federated learning was limited, and thus, the scalability of the framework was hindered and the number of devices that can be tested at the same time was limited. It was also important to guarantee that the models were converging in the same way each federated round, especially in the case of skewed data or different capabilities for each device. Hardware availability also restrained experiments to reduce the scope of energy consumption analysis and long term deployment tests. Additionally, while secure aggregation and other security mechanisms were discussed, their implementation was not fully evaluated. These limitations point toward future improvement, possible to validate and strengthen the framework by real-world application.

6.5.2 Future Research Directions

This study serves as a stepping stone towards large-scale smart farming, and possible future directions involve improving the scalability and flexibility of the proposed Federated Learning and Edge Computing framework. For example, there is a need for lightweight FL models for FWIoT devices with limited resources to effectively train FL models with high accuracy. Research on more sophisticated privacy-preserving methods, such as homomorphic encryption and secure multi-party computation, can also be extended to further increase the privacy of the actual data in resilient model updates. Another promising direction is integrating blockchain technology with the current models, so that transparency and trustworthiness can be made possible in the model aggregations. On the other hand, conducting evaluations in semi–open world scenarios, where different farm environments with diverse sensor data are tested/accounted for, will yield further information about the robustness and performance of the system. In addition, a more powerful extension of the existing implementation would enable models to learn dynamically, adjusting to changes in environmental conditions or sensor inputs in real-time, thereby enhancing the accuracy of decision making. Lastly, a comprehensive energy consumption analysis and sustainability analysis reading to long-term viability for practical crop production could be incorporated.

6.5.3 Potential for Real-World Deployment and Commercialization

This combination demonstrates great potential for real-world implementation and commercialization in agriculture. By enabling the processing of data on edge devices, it significantly decreases the latency and cost of bandwidth which are required for remote farms to operate efficiently with limited connectivity. The framework is designed to improve data privacy and security by ensuring that sensitive farm data does not leave the local environment by default, which complies with many new and emerging data protection regulations. Its design is compatible with commonly used platforms like AWS IoT Greengrass and TensorFlow Federated, thus allowing for easy integration with existing agricultural infrastructure. Also the framework is adaptable to various smart farming applications including automated irrigation, pest control and crop health monitoring. This gives room for agri-tech startups, IoT solution providers, and agricultural service platforms to monetise this technology. The solution can provide scalable, efficient and secure smart farming on various agricultural environments with further refinement and testing.

7 Conclusion and Future Work

This study was carried out to improve IoT designed smart farming through federated learning and edge computing. The architecture was designed using TensorFlow Federated and AWS IoT Greengrass for low latency, privacy preserving turning of the sensor data. Data being processed on edge devices locally and only model update sent to cloud, reducing bandwidth and protecting privacy. This approach gained a latency reduction of over 45%, a staggering improvement in the context of creating and deploying new machine learning techniques such as federated learning. Not only does this staggering reduction enhance the responsiveness of real-time applications such as smart farming, but it also indicates the potential of edge computing for latency-sensitive applications. With further research into model architecture and more advanced hyperparameter optimization techniques, even greater performance and efficiency gains are anticipated. Although the results were promising, utilization of simulated datasets and resource constraints on the edge devices hindered scalability and realism of deployment. They also touched on security components like secure aggregation, but didn't build those out. The future directions comprise of developing a lightweight model for constrained edge devices, incorporating blockchain to enhance model aggregation transparency and evaluate the framework across various real-world farm settings. The solution proposed has high commercial potential as stand-alone approach for most of the key pain points in the agricultural domain, the responsive, scalable, and secure nature of the data processing from IoT parameters, which can subsequently translate the processed parameters into actionable responses such as automated irrigation, pest control, and crop monitoring in different types of farms, using their own systems, further adds value to the solution.
Furthermore, the architecture is feasible to implement in the real world using widely available technologies such as AWS IoT Greengrass and TensorFlow Federated, with very few changes to existing farm infrastructure. Its edge-based distributed architecture also ensures horizontal scalability, so that it would be able to manage sensor networks effectively in smallholder as well as large-scale farms, thereby enabling broader adoption and long-term operational sustainability.

References

Abu-Khadrah, A., Ali, A. M. and Jarrah, M. (2023). An amendable multi-function control method using federated learning for smart sensors in agricultural production improvements, *ACM Transactions on Sensor Networks* .

Akkem, Y., Biswas, S. and Varanasi, A. (2023). Smart farming using artificial intelligence: A review, *Engineering Applications of Artificial Intelligence* **120**: 105899.

Dineva, K. and Atanasova, T. (2022). Cloud data-driven intelligent monitoring system for interactive smart farming, *Sensors* **22**(17): 6566.

Friha, O., Ferrag, M. A., Shu, L., Maglaras, L., Choo, K.-K. R. and Nafaa, M. (2022). Felids: Federated learning-based intrusion detection system for agricultural internet of things, *Journal of Parallel and Distributed Computing* **165**: 17–31.

Gopidesi, C. (2023). Smart agriculture dataset, `https://www.kaggle.com/datasets/chaitanyagopidesi/smart-agriculture-dataset`. Accessed: 2025-04-17.

Halai, N. (2022). Crop recommendation dataset, `https://www.kaggle.com/code/niteshhalai/crop-recommendation-dataset`. Accessed: 2025-04-17.

Kalyani, Y. and Collier, R. (2021). A systematic survey on the role of cloud, fog, and edge computing combination in smart agriculture, *Sensors* **21**(17): 5922.

Kondaveeti, H., Sai, G., Athar, S., Vatsavayi, V., Mitra, A. and Ananthachari, P. (2024). Federated learning for smart agriculture: Challenges and opportunities, *2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE)*, IEEE, pp. 1–7.

Kumar, P., Gupta, G. P. and Tripathi, R. (2021). Pefl: Deep privacy-encoding-based federated learning framework for smart agriculture, *IEEE Micro* **42**(1): 33–40.

Le, D., Dao, M., Tran, A., Nguyen, T. and Le-Thi, H. (2023). Federated learning in smart agriculture: An overview, *2023 15th International Conference on Knowledge and Systems Engineering (KSE)*, IEEE, pp. 1–4.

Manogaran, G., Alazab, M., Muhammad, K. and De Albuquerque, V. (2021). Smart sensing based functional control for reducing uncertainties in agricultural farm data analysis, *IEEE Sensors Journal* **21**(16): 17469–17478.

O'Grady, M., Langton, D. and O'Hare, G. (2019). Edge computing: A tractable model for smart agriculture?, *Artificial Intelligence in Agriculture* **3**: 42–51.

Patil, P. (2022). *Smart Farming IoT Sensor Data Filtering Using Pattern Analysis and Edge Computing to Reduce Latency*, PhD thesis, National College of Ireland, Dublin.

Qazi, S., Khawaja, B. A. and Farooq, Q. U. (2022). Iot-equipped and ai-enabled next generation smart agriculture: A critical review, current challenges and future trends, *IEEE Access* **10**: 21219–21235.

Shaikh, T., Rasool, T. and Lone, F. (2022). Towards leveraging the role of machine learning and artificial intelligence in precision agriculture and smart farming, *Computers and Electronics in Agriculture* **198**: 107119.

Wisam1985 (2024). Iot agriculture 2024 dataset, `https://www.kaggle.com/datasets/wisam1985/iot-agriculture-2024`. Accessed: 2025-04-17.

Wolfert, S., Ge, L., Verdouw, C. and Bogaardt, M.-J. (2017). Big data in smart farming – a review, *Agricultural Systems* **153**: 69–80.