Corn Yield Prediction Using Crop Growth and Machine Learning Models

Audrey Moswa

Thesis submitted to the University of Ottawa
in partial Fulfillment of the requirements for the
Master of Science in
Computer Science

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

Undoubtedly, the advancement of IoT technology has created a plethora of new applications and a growing number of devices connected to the internet. Among these developments emerged the novel concept of smart farming. In this context, sensor nodes are used in farms to help farmers acquire deeper insight into the environmental factors affecting their productivity.

In recent years, we have witnessed an emerging trend of scholarly literature focused on smart farming. Some focus has been on system architecture for monitoring purposes, while another area of interest includes yield prediction. Humidity, air and soil temperature, solar radiation, and wind speed are some key weather elements monitored in smart farms.

We introduce a mechanistic crop growth model to predict crop growth and subsequent yield, subject to weather, soil parameters, crop characteristics and management practices. We also seek to measure the influence of nitrogen on yield throughout the growing season. The machine learning models are trained to emulate the crop growth model in the state of Iowa (US). The multilayer perceptron (MLP) is chosen to evaluate the model prediction as it generates fewer errors.

Furthermore, the MLP optimization model is used to maximize corn yield. The experiment was performed using different scenarios, stochastic gradient descent (SGD), and adaptive moment estimation (Adam) optimizers. The experiment results revealed that the SGD optimizer and the dataset with the scenario of unchanged parameters provided the highest crop yield compared to the mechanistic crop growth model.

Acknowledgements

First and foremost, I thank the Almighty God, through whom I had the knowledge, wisdom and strength to write this thesis.

I would like to express my sincere gratitude to my thesis supervisor, Dr. Iluju Kiringa and co-supervisor, Dr. Tet Yeap, for their assistance, guidance and encouragement to complete this thesis. Without their support, I would have never succeeded in achieving this milestone.

Special thanks goes to my family for their continuous prayers, motivation and reassurance throughout my study.

I would also like to thank my colleagues and friends for their tremendous support and advice.

Dedication

In memory of my late father, Prof. Dr. Joseph Lokonda MOSWA, for his huge sacrifices. It is his outstanding qualities that I try to emulate in all I do.

I dedicate this work to my family for their continuous prayers and unparalleled love and support, making me who I am.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The Internet of Things (IoT) refers to a network of systems designed to communicate, perform distributed sensing and compute in collaboration with other devices. This sensing allows it to collect real time data, which can later be used to improve different models and help in prediction [4].

Due to large amounts of generated data, new data collection and processing techniques need to be designed with the corresponding infrastructure to handle this extensive data. Some uses of these IoT devices include smart cities, industry, and agriculture [5][6][7][9]. In one of the critical application areas mentioned, precision agriculture is of great importance. Some existing challenges include the efficient use of inputs such as water, fertilizer, and even labour.

Due to the global increase of population and food shortages, the main goal of precision agriculture is to increase production and maintain food costs affordably.

Records show that in 2019, corn production exceeded that of all other crops in the US [104]. Now that replacing gas in cars with ethanol has gained traction, it is essential to enhance the quantity of corn production. Therefore, predicting corn yield could provide relevant insight for decision-making [2].

Many studies have used crop models for cropping systems and predicting applications, such as mechanistic and machine learning models. Results demonstrated their capabilities and significant advancement in yield prediction in agriculture [104].

The mechanistic crop growth model establishes a relationship between input and output variables that is mechanistic in nature and applicable to crop management, pest control and yield prediction [113]. The mechanistic model describes corn growth and yield in

response to the essential factors that influence them, such as weather, environment, and nutrient inputs [114].

Machine learning has become an integrative feature of modern techniques, suggesting automated processes for predicting phenomena based on past observations, discovering underlying patterns in data and providing insight into problems [113][114]. In addition, ML predictive models enable the optimization of dataset variables.

The present work uses an IoT-based smart agriculture system architecture for corn yield prediction and suggests machine learning algorithms to run in the architecture.

The present work then proposes a mechanistic crop growth model to predict corn growth and yield in a US state (Iowa) using historical data from 1982. In addition, it suggests machine learning (ML) models for optimizing dataset parameters from the mechanistic crop growth model.

The precipitation and nitrogen nutrient the corn receives throughout the growing season strongly impacts the development and growth of the crop.

1.1 Motivation

The Agricultural sector faces food scarcity and insufficient cost-effectiveness due to the global increase of population. The proposed remedy to this situation is to introduce smart technologies to meet the world's food demands for the coming years.

Studies show that advancements in machine learning and crop modelling have created the potential to maximize prediction in agriculture. These techniques have significantly improved prediction performance; however, they have been mostly evaluated individually. Coupling them may further enhance corn growth and yield prediction and enable optimization.

Since the mechanistic model is non-continuous and contains conditional statements, it should be transformed into a continuous non-linear function to optimize the variables. Machine learning models such as multilayer perceptron (MLP) are used to optimize different factors that impact corn growth prediction.

Maize (Zea mays L.) is an important crop and used for many purposes such as food, livestock feed, fuel and industrial products. In past years, the increase of maize yields were associated with increased amounts of nitrogen (N) fertilizer application to crops. Nitrogen plays a crucial role in crop metabolism, and an adequate supply of soil nitrogen is critical to increasing crop yield.

The combination of machine learning and mechanistic crop growth models is expected to allow precision agriculture to deliver high operational efficiency, maximize crop growth and yield, reduce production costs, and efficiently utilize inputs (water, fertilizers, etc.) [4].

1.2 Problem Description

A key objective in the agricultural domain is to increase productivity. The challenge is to efficiently use inputs that influence production, such as water and fertilizer.

To achieve this goal, it is required to use advanced techniques that can assist in managing these factors. Because the agricultural arena is an essential source of data, managing and investigating the generated information is integral to better decision-making and predictions.

The data collected from various fields should be investigated to analyze processes like crop improvement, yield prediction, or water stress identification in order to increase productivity.
Crop yield prediction is among the primary processes involved in crop growth, and the mechanistic crop growth model will assist in predicting crop growth and yield in order to improve production.

After assessing the crop yield prediction, we can apply different optimization algorithms to factors that affect crop growth to maximize yield prediction.

Another technique useful for yield prediction is machine learning. Its various predictive models are utilized to optimize the input values and analyze the results. These methods enable an analysis of significant factors involved in crop growth.

1.3 Solution Outline

We used a three-tier system architecture including field device layer, fog/edge computing and cloud computing layer. The architecture will perform data analysis using machine learning methods to enhance predicted events and make more reliable decisions. The architecture engages in the collection, transmission, and operation of physical parameters that influence the farming field such as weather, air and soil temperature, soil moisture levels, relative humidity, and radiation. Such information is then used to efficiently manage crop production.

We proposed a mechanistic crop growth model combined with machine learning to predict and optimize corn growth and yield. We assessed the model to predict corn yield accurately based on historical data. We then varied the applied N fertilizer values and determined the appropriate N value we could apply to get the maximum yield. Then, we used machine learning methods to optimize the variables.

We selected two predictive ML models; namely, the random forest regressor (RF) and multilayer perceptron (MLP), and trained them using the output of the mechanistic growth model. Then, we compared the models using error metrics, such as the root mean square error (RMSE), relative root mean square error (RRMSE), mean absolute error (MAE) and coefficient of determination (R^2). We considered the model that provided the most accuracy and least prediction error.

1.4 Methodology

In this thesis, we based our research on predicting and optimizing corn growth and yield. We first gathered data from USDA-NASS and proposed a mechanistic model of crop growth to accurately predict the growth of corn and its subsequent yield.

We then combined the yield results of the mechanistic corn growth model with the machine learning models (MLP and random forest). We evaluated their performance to improve the corn growth and yield prediction in smart farming.

In addition, we varied the applied N fertilizer values to estimate the amount that maximizes the corn yield and used optimization algorithms to optimize the crop yield. Finally, we compared crop yield results obtained from the mechanistic crop growth and MLP models.

1.5 Contribution

This thesis provides the following contributions:

- We put forth a mechanistic crop growth model to assess corn growth and yield based on historical datasets from the USDA-NASS. We listed the different mechanistic crop growth models and investigated the advantages and disadvantages of the related work. We simulated the mechanistic crop growth model to predict the biomass accumulation, corn growth and yield. We then monitored the grain yield variation when the applied N fertilizer amount varied.

- We combined the output from the mechanistic crop growth model with machine learning models, including MLP and random forest models. Furthermore, we applied optimization models such as stochastic gradient descent (SGD) and adaptive moment estimation (Adam) to the dataset variables. We performed some scenarios to maximize corn growth and yield prediction. The simulation results revealed that the SGD optimization model with the dataset with unchanged parameters provided the highest crop yield.

1.6 Outline of Thesis

This thesis is structured as follows: Chapter 2 details the necessary background and literature review of concepts related to IoT, precision agriculture, and smart farming, and it also details the three-tier system architecture proposed. Chapter 3 explains the yield prediction using the mechanistic crop growth model and machine learning. Chapter 4 presents all simulation experiments and summarizes results, and finally, chapter 5 concludes our work and discusses future work.

Chapter 2

Background and Related Work

This chapter presents details on the IoT paradigm, its architecture, and its uses in precision agriculture and smart farming. We also discuss different crop growth models. Finally, we describe machine learning models and their use in agriculture.

2.1 Internet of Things

The Internet of Things has found its application in many domains, such as smart city, industry, healthcare and agriculture. The primary objective of IoT is to integrate the physical world and the virtual world to communicate and exchange information through the internet. The main advantages of using IoT in agriculture are achieving higher crop yields and less cost. IoT systems are designed to perform distributed sensing, computing and communication with other devices in a collaborative manner. This sensing allows the system to collect real-time data which can later be used to improve different models. . In [5][22][23], the IoT system consists of five different layers: IoT sensing, communication network, storage and processing, learning, and application. The communication between these layers is established by integrating several heterogeneous IoT devices and enabling clients to access and control the IoT devices.

2.1.1 IoT Abstraction Layer

According to [23][49], IoT can be divided into five abstraction layers: IoT sensing, communication network, storage, learning, and application. These layers are expected to in-

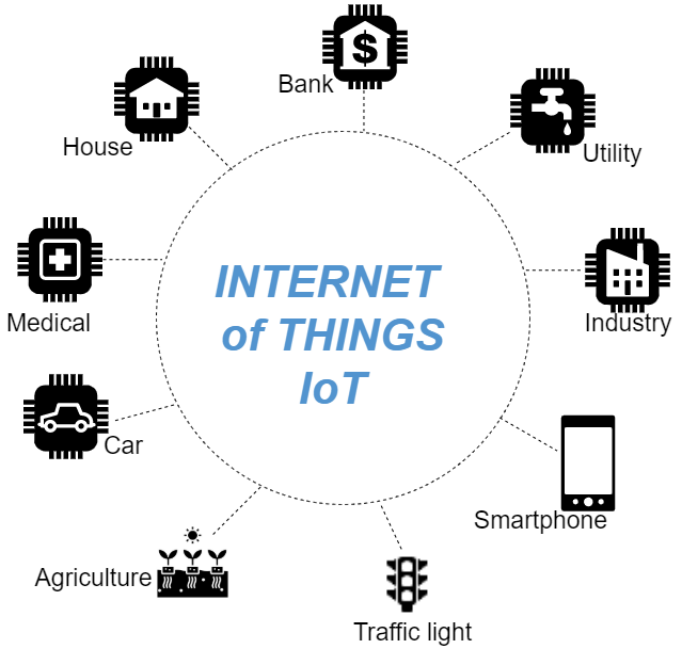


Figure 2.1: Internet of Things

corporate various IoT devices to allow communication between them and provide clients with access and control to IoT devices.

2.1.1.1 IoT Sensing

The IoT sensing layer comprises IoT devices, including sensors and actuators, that collect generated data from a specific environment and interact with it. The operations that can be performed in this layer are data acquisition or sensor selection [59].

2.1.1.2 Communication Network Layer

This layer plays a crucial role in the deployment of IoT systems. It enables communication between multiple low-level nodes and the edge node, fog node, and cloud [23][24]. It is also responsible for aggregating information from existing technologies. The communication layer manages all constraint requirements of the nodes. A few machines use existing technologies such as machine-to-machine communication to allow resources to implement full-fledged networking protocols [49]. IoT middleware and connectivity protocols

are developed in this layer to allow connectivity between heterogeneous systems [84].

2.1.1.3 Storage Layer

The storage layer collects and stores various data such as sensor data, aggregated data and knowledge discovery. According to their complexity, there are several strategies to store data for placement in a specific location [60]. For example, critical applications that require managing end-to-end IoT architecture can be stored in the edge/fog, and the cloud may store aggregated data. In addition, IoT devices and gateways can perform computation and analysis to reduce latency and improve the quality of service (QoS) [5].

2.1.1.4 Learning Layer

The learning layer consists of analyzing collected sensor data. Once data is stored, the layer performs a comprehensive data analysis by extracting knowledge from raw data and managing where the system's intelligence is performed. To manage the target system's requirements, this layer attempts to optimize data analytics location [25].

2.1.1.5 Application Layer

The application layer consists of communication protocols that interface the IoT system and devices. It provides users with an environment to monitor data processed through a web server or an android application, and data are received and stored securely. The application layer monitors data and provides intelligent suggestions to improve the system and store it for further analysis [26].

2.2 Big Data

In the past few years, the rate of data generated has increased tremendously, making the traditional database paradigm insufficient to store the generated data. One of the alternatives to this paradigm is the use of cloud storage [45]. The Big Data approach refers to collecting a massive amount of heterogeneous data from multiple sources and more extended periods [61]. The main difficulty is to capture, store, analyze, and search from the data. These steps are required for data processing and big data analytics to discover hidden patterns and optimize predictions from the data [46]. In precision agriculture, big data is used to improve productivity and supply chain management [4][47][48].

2.2.1 Big Data Analytics and Processing

There exist several technologies in big data, such as Hadoop [70], which is developed for big data analytics and processing [46]. Hadoop consists of different components, including HBase for structured and unstructured data storage and management, Drill and Storm for interactive analysis and stream processing, Hadoop Distributed File System (HDFS) for storing massive amounts of data and MapReduce for parallel and distributed processing of data [45][46].

2.2.2 Characteristics of Big Data

The characteristics of big data consist of five V's: volume, velocity, variety, veracity and value [62]. The main features are reflected in the first three. Volume is the quantity of data generated and is currently measured from petabytes to zettabytes. Velocity refers to the frequency and speed at which data is generated, captured, and shared. Variety represents the different types of data sources based on content, location or geospatial. In addition, variety consists of structured and unstructured data [47].

2.3 Cloud Computing

Cloud computing allows sharing of platforms, resources, and networks for various applications at an economical cost. It is used to store data from heterogeneous sources including, IoT devices [48]. Cloud computing proposes three different services including software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS), also referred to as hardware as a service (HaaS) [49][63]. SaaS provides users with access to different applications over the internet. PaaS provides users with the flexibility of managing and maintaining infrastructures of applications without building them from scratch. IaaS or HaaS provides computing infrastructure to improve performance operations. It manages storage, virtual machines and networking to users [49].

Recently, the number of IoT devices connected to the network has increased and generates a large amount of data, the deployment of which requires location awareness and available bandwidth. Cloud computing models become limited in their ability to manage produced data and real-time requirements for IoT applications.

Cloud computing requirements such as costs associated with centralized processing and latency become challenging for the cloud computer model [40][83]. Some cloud service providers for IoT applications are Microsoft Azure, Google Cloud Platform, Amazon

Web Services, and IBM Cloud. The cloud computing model uses a distributed database management system (DDBMS).

2.4 Fog Computing

The notion of fog computing was first proposed by Cisco [50]. Fog computing as a concept initially emerged as a computing organization alternative to leverage intelligent network edge devices that make up modern IoT systems [24]. Fog computing is strictly linked to IoT, which expands the cloud and some services close to the data source to avoid excessive cloud resource exploitation. Fog computing aims to move processing abilities closer to end-users, avoiding excessive exploitation of cloud resources, further reducing computational loads, and minimizing latency. Fog computing is a layer between the cloud and edge nodes [49]. The fog has been presented as complementary to conventional centralized systems; however, fog cannot be considered as another type of cloud computing and cannot replace it [51].

Fog computing has some limitations related to computational complexity, storage and reliability of network infrastructure as part of the overall architecture. Also, fog computing is not always connected to fixed and reliable network infrastructure [39][40][41]. Some disadvantages of fog computing in smart farming are explained as follows:
- Security: Fog computing experiences issues due to the attack of malicious users on gateways or when non-authenticated users enter the network and try to thieve critical information. When data is transferred back and forth between the sensors located in the field and the gateway, the architecture becomes vulnerable.
- Complexity: Fog computing places intelligence at the local area network (LAN), which increases the complexity of the system. In the fog computing environment, there are a large number of fog nodes, which can thus increase complexity.
- Privacy: Fog nodes are easily accessible to anyone due to their presence on edge; sometimes, an unauthorized person can try to modify the data.

2.5 Edge Computing

The idea behind edge computing is to extend cloud computing capabilities to the network's edge to push computation, networking, and storage closer to the proximity of data sources [37][39][40]. This significantly reduces the amount of data sent to the network and the bandwidth needed to generate them. When it comes to making a critical real-time

decision, such as irrigation control, edge computing significantly reduces the latency of the information travelling to the cloud and then back.

The difference between edge computing and fog computing is that the fog computing layer shifts the edge devices' tasks to gateways that are connected to the communication network and are physically distant from the IoT devices [49]. The implementation of the edge layer can be performed in three modes: mobile edge computing, fog computing, and cloudlet computing [49][51].

2.6 Gateway

Gateway is an intermediate component that operates as a communication medium between sensor nodes and the cloud [23,60]. It is responsible for gathering heterogeneous sensor data, aggregating data, performing lightweight data processing, and analytics. The gateway consists of a local database to temporarily store the sensor data before forwarding them to the cloud. In many general-purpose proposed IoT architectures, all the data generated by the things are transmitted to the cloud via the gateway or fog nodes located in the LAN network [81][82].

This approach has implications for the real-time applications of the platform. It causes high propagation latency and results in under utilization of the gateways [23]. A well-known example of the gateway hardware in IoT is the Raspberry Pi. The gateway supports different communication protocols such as ZigBee or LoRa to maintain communication with the sensor nodes and has an internet connection via Wi-Fi, Ethernet, or LTE [24].

2.6.1 Communication Interface

The sensors' output in [6] is read by an Arduino Uno connected to Zigbee for sending data to a gateway node that will aggregate the received data and store it locally in SQLite and send the data to the server using web service. The web service for field sensor data collection is written in PHP with a lightweight REST API to communicate the data between the field device and the server.

A web service for online weather data collection has been developed in Python to collect weather forecasting data like temperature. Web forecasting portals such as AccuWeather use API and provide the information in JSON, XML, or HTML format. The developed web service uses API technology to read forecasted data for a specific location, which is stored

in the MySQL database server. This is taken into account in the prediction algorithm. However, the limitation of [6] is that the analysis was not done in real-time.

2.6.2 Data Analysis

In [7], an approach based on data analysis and processing is presented to monitor soil and crops where stratified accumulation and modeling primitives influence the strength of the network and reduce the energy needed for unnecessary data transmission. The proposed framework relates how the two-tier fog computing system can greatly decrease the quantity of data transmission to the cloud and enhance computational load balancing, thus reducing latency.

Among the methods that were used to derive top-level information from the sensor data was the symbolic aggregate approximation (SAX) approach [65][66]. This approach works to allocate symbols to time series segments and adapt them in a cohesive manner. The SAX approach is part of a group of time-series data mining methods connected to non-parametric modeling.
In this study, they used three interpolation techniques: linear interpolation, cubic spline, and piecewise cubic Hermite interpolating polynomial. The limitation is that no ML techniques were used, and interpolation techniques were applied to reconstruct the data when it was transferred to the cloud layer.

2.7 Microcontroller

Microcontrollers use network interfaces to communicate with IoT devices, such as sensors or actuators [67]. They push sensor data to the gateway for any analysis. The microcontroller is made up of hardware and software elements tasked with transferring sensor data to the gateway through wireless communication technology. It also generates signals or actions for the actuators to perform [68]. The design of microcontrollers can support one or many network protocols such as Wi-Fi, Bluetooth and cellular networks like 2G/3G or even RFID.

2.8 IoT-based Smart Farming

As a new application, smart farming serves to advance IoT technology. It supports farmers in making smart decisions by rendering agriculture more connected and intelligent

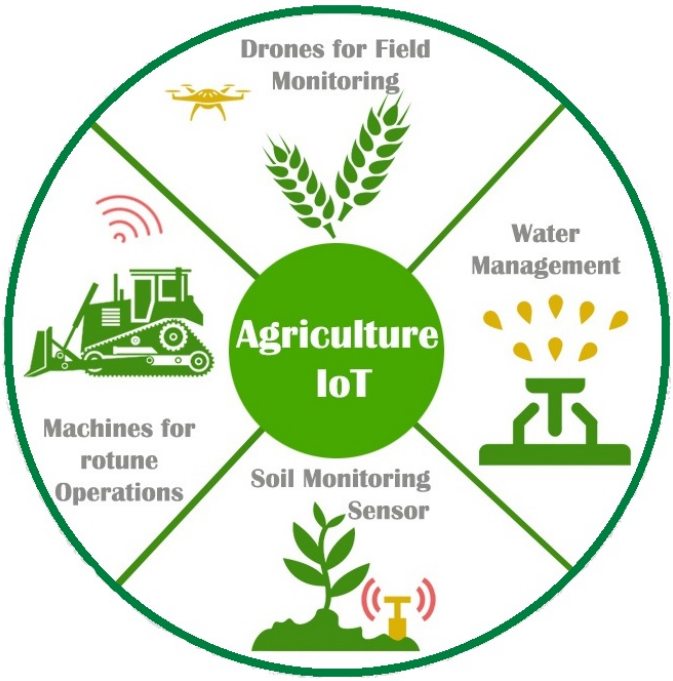


Figure 2.2: Internet of Things [69]

[71][72][73]. It also reduces the use of critical resources and improves productivity while lowering farming costs [69].

2.8.1 IoT Architecture in Smart Farming

IoT architecture requires concurrent data collection to support, analyze and control data from various sources. It also needs good connectivity and communication between devices [75]. Additionally, it requires robust protocols between sensors/actuators and the cloud, availability, and high quality service [42][74]. IoT architecture can be simplified into three fundamental layers [74]:

1. Perception Layer - this is composed of the embedded systems and edge devices that interact with the environment. It performs sensing, lightweight storage, and machine learning.

2. Network Layer - processes data and provides more heavy-duty storage and machine learning in comparison to the perception layer.

3. The Application Layer - performs data processing and more heavy-duty storage and

machine learning compared to the perception layer.

Below is detailed the IoT environment in the agricultural context based on four main components which are:

2.8.1.1 IoT Devices

IoT devices are comprised of embedded systems that interact with sensors and actuators and require a wireless connection [82][104]. They interface and communicate with the cloud via communication technology through gateways and measure in-field parameters including air temperature and solar radiation.

2.8.1.2 Communication Technology

Communication technology represents a crucial role in the execution of IoT systems. There are many available communication technologies, which can be categorized into two groups: short-range connections and long-range connections. In terms of short-range connections, the communication network uses ZigBee, an IEEE 802.15.4 standard, to connect sensors [77][78]. Networks use a mesh topology for collecting data from sensors and mostly operate in the 2.4 GHz band, and sometimes in the 868/915 MHz bands. According to the environment's features, Zigbee can cover distances ranging from a few meters up to 100 meters [24]. However, the communication between sensors in the field and fog nodes is done by long range (LoRa).

The main objective of the LoRa protocol is interoperability between layers. This wireless technology uses licensed broadband cellular standards, named low-power wide-area networks (LPWANs). In LoRa technology, a star topology is employed, enabling long-range communication, low bit rate and low power consumption [76]. The features of LoRa make it more appropriate for IoT applications. This technology allows one gateway to support large fields by exploiting the star topology, and battery life for the nodes is typically very long, lasting up to 10 years [24][37].

Message Queuing Telemetry Transport (MQTT) Protocol - Communication between fog nodes and the cloud is facilitated by the MQTT protocol, a machine-to-machine (M2M) protocol with few message types and smaller message sizes. MQTT can send control instructions to fog nodes through an encrypted data transmission mode. MQTT is a messaging protocol that follows a publish/subscribe architecture.

It consists of three different components, namely, publisher, broker and subscriber, which support connections from edge nodes to the cloud with limited computing power. It is thus most suited for resource-constrained environments (low-speed wireless access). The publisher detects data and publishes to the broker for processing and analyzing purposes, entering sleep mode when not in active use. The broker then forwards the information to the subscriber. Whenever the broker receives new data, it will inform the subscriber [7][24][38].

2.8.1.3 Internet

The Internet forms the core network layer that discovers, translates, and exchanges data and network information between different subnetworks [7][49]. The internet enables connection between IoT devices and makes data available anywhere and anytime.

2.8.1.4 Data Storage and Processing Units

Data-driven agriculture requires the collection of a large amount of data from various sources, which need to be stored and processed. The use of cloud IoT platforms allow big data collected from sensors to be stored in the cloud. Lately, edge and fog computing paradigms have been proposed to reduce costs and latency for critical applications, in addition to improving computational load balancing and QoS [5].

Edge Computing - Assumes a clustering approach where each cluster has a cluster head, which can be a router or hub. After collecting information on the agriculture field area, intelligent edge devices process the data locally [3][4][7]. This significantly reduces the amount of data sent to the network and the bandwidth needed to generate them. When it comes to making critical real-time decisions, such as irrigation control, edge computing significantly reduces the latency of information travelling to the cloud and back.

Fog Computing - The aim of fog computing is to move processing abilities closer to end-users, avoiding excessive exploitation of cloud resources and further reducing computational loads [7]. Fog computing is located between the cloud and IoT devices. Fog nodes mainly deal with the cluster heads, collecting data from them and performing data processing primitives for intelligent aggregation to significantly reduce the amount of data transmitted to the Cloud. This serves to improve computational load balancing and reduce wait times. The difference between edge computing and fog computing is that the fog computing layer

shifts the edge device's tasks to gateways that are connected to the communication network and are physically distant from the actuators and the sensors [5][6].

Cloud Computing : Data is sent to the cloud platform through fog nodes, which are connected to the Internet via a particular communication network [7]. After receiving the perceived data, the cloud processes and stores the data and makes decisions on the data values according to trigger conditions.

2.9 IoT-based Architecture

This section presents an IoT-based architecture planned to support crop growth models. The architecture is illustrated in figure 2.3 and is a revised version of the architecture proposed in [7].

2.9.1 System Model Architecture

The architecture is a combination of IoT and machine learning techniques. It has been designed for efficient data collection and processing to improve production.

It is divided into three physical layers: cloud computing, fog computing, and edge computing or field devices [39].

The idea behind fog and edge computing is to extend cloud computing capabilities to the network edge to push computation, networking, and storage to the proximity of data sources [40][41]. In addition, we suggest various machine learning algorithms to be run in different architecture layers to predict corn yield accurately.

1. The field devices layer contains different types of nodes, including sensors, actuators, and microcontrollers. The sensor nodes might be a standalone node or a wireless sensor network and consist of soil moisture sensors, humidity sensors, solar radiation sensors and soil temperature sensors. The data from sensors is sent via Zigbee or Wi-Fi to the microcontrollers to be read and transmitted to the gateway [78].

The gateway will aggregate the data and store a part of it in a local embedded database, and an ML prediction algorithm will analyze them and predict the crop yield based on the sensor data [6].

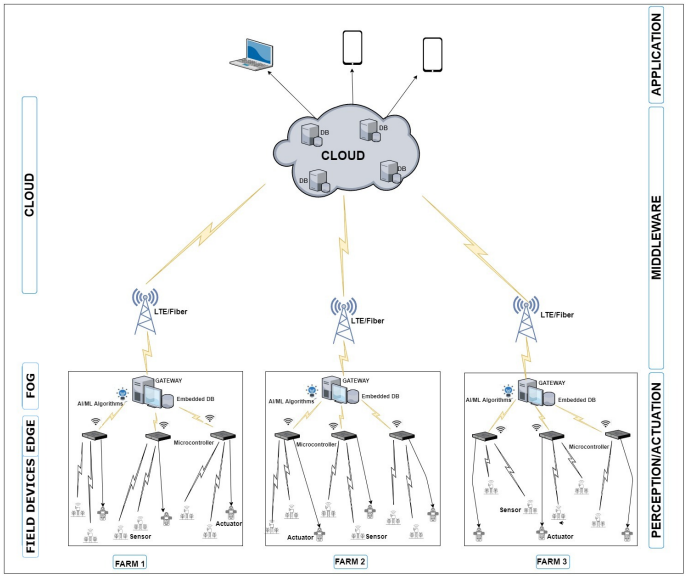


Figure 2.3: Architecture of a Smart Farm

2. After aggregating data, the proposed ML algorithm will analyze the data (soil temperature, soil moisture, etc.) and predict those locally stored based on the field sensor's data and the weather forecast. The gateway transmits the other part of the aggregate data directly to the cloud via the internet, using a particular communication network (LTE, Fiber) [65].

After receiving the perceived data, the cloud processes and stores the data to make decisions on the data values according to trigger conditions.

2.9.2 Architecture Layers

2.9.2.1 Perception Layer

The perception layer is the lower layer of the architecture that is close to the data sources. It consists of edge and field devices (sensors, actuators, and microcontroller), which perform sensing, lightweight storage, and interact with each other. The purpose of the edge is to bring the intelligence and processing capability closer to the data source, while the fog serves to perform networking and machine learning.

For real-time data analytics and decision-making, the perception layer needs to avoid latency and have embedded data storage in case all data is sent to the cloud. The perception layer connects to the middleware layer through a wireless network such as Wi-Fi, LTE, GSM or Bluetooth [5]. In this architecture, The MQTT is used as a messaging protocol for communicating between the perception layer and the middleware layer since it is lightweight and designed for low bandwidth, high latency systems and unreliable networks [38].

2.9.2.2 Middleware Layer

The middleware layer exists between the perception layer and the application layer. It provides abstraction in the cloud, which, contrary to the perception layer, performs significant data storage, networking, service management and machine learning.

The cloud serves as an essential component of this layer to store and analyze substantial perceived data [3][5]. The function of this layer is that of a bridge between the IoT devices and the applications as it interfaces both the perception layer through MQTT [38] and the application layer through HTTP protocol.

2.9.2.3 Application Layer

The application layer represents the uppermost layer of this system. It works to interface with the farm system and offers tools for crop management and yield prediction. This layer provides specific services for users based on processed and analyzed data. Also, it provides the farmer access to data through a particular interface [60].

2.10 Precision Agriculture and Smart Farming

Precision agriculture (PA) is a farm management method that uses IT and specialized equipment (sensors, data analysis tools) to enhance productivity and decision-making in agriculture [44]. PA's main objective is to support farmers in managing their business by optimizing the inputs and reducing the environmental impacts to ensure profitability, sustainability, and environmental protection [42]. PA uses data generated by multiple sources to enhance crop productivity, yield and quality. It also increases the efficiency of crop management procedures such as fertilizer administration, irrigation processes, and pesticide applications. PA has access to real-time data from the field regarding the condition of crops and other factors, including temperature, soil, and weather forecasting.

Real-time data is provided from various sources and then transmitted and analyzed using communication technologies and artificial intelligence methods. The implementation of IoT in precision agriculture has a two-fold purpose—sensing data from the field and analyzing data to assess the necessary response. IoT implementation in PA requires sensor integration, automatic control, information processing, and network connection. PA uses machine learning algorithms to explore the data and predict subsequent actions based on the experience learned from them; then, data analysis is performed to achieve operational effectiveness [5].

Unlike PA, smart farming [4][48][72][73] is the efficient use of information and communication technology such as IoT and cloud computing to manage and optimize complex farming systems. A variety of data is collected from smart machines located on the field and is processed and analyzed to provide farmers with access to concrete data to help decision-making [79, 80].

2.10.1 Data Collection

There are various technologies for data collection in precision agriculture, such as remote sensing, satellites, and unmanned aerial vehicles (UAV).

Remote Sensing: wireless sensor networks or standalone sensors are used to measure and collect data from the physical environment. Sensors measure parameters such as soil moisture, temperature, and pH level, which are then sent through a gateway to be stored in the cloud.

Satellites and UAVs capture real-time aerial imagery and estimate data related to various soil properties, plant health, and crop yields. This helps to avoid wasting resources, reduce costs and control the farm's environmental impact [43].

2.10.2 Mechanistic Crop Growth Model

The monitoring of crop growth and yield prediction is one of the most critical processes to ensure food availability and promote sustainable agriculture. The development of crop mechanism research, photosynthesis measurement technology, and computer technology has advanced crop growth simulation research and enabled the creation of a significant number of crop growth models. Essentially, crop growth models are used to understand

the crop production process, analyze strategies, and manage agricultural systems [30]. Crop growth models are now used to aid in decision-making for erosion control, climate fluctuations, water and fertilizer use, and yield forecasting. Crop growth models can also project crop performance in areas where the crop was not previously planted or where conditions were not optimal for growth. Estimating agricultural output based on weather and soil conditions as well as crop management is a key purpose of crop simulation models [31].

Based on physiological and ecological mechanisms and relating crops' growth with the physical system, crop growth models have been among the most potent agricultural research tools [32]. A simple water balance proves that there is an existing relationship between water and crop yield. The relevance of crop growth simulation has driven many scientists to build crop growth models for crops. Some successful models used to simulate maize growth and output are the decision support system for agrotechnology transfer (DSSAT) [57], crop-environment resource synthesis (CERES)-maize [10], and the erosion-productivity impact calculator (EPIC) [33]. Contrary to empirical crop production, the simulation approach reduces the time for execution. Most growth models evaluate progress by employing a daily time-step, and few require hourly time-steps.

Generally, a crop model can replicate the impact of weather, phenology, soil water, photosynthesis and nitrogen on the growth and output for a specific crop. As shown in figure 4.1, the crop model requires minimum weather data, including maximum and minimum air temperature, daily solar radiation values, and precipitation. Factors such as humidity and wind speed are optional data. Soil data depends on a water balance that includes soil evaporation, drainage coefficient, runoff curve number, soil water, nutrient, and temperature.
The crop model result includes phenological events, growth details, soil temperature, moisture, soil nutrients, and the plant daily. Crop models feature different characteristics and can be mainly based on extensive assembly of empirical functions for operations involved in plant growth, such as the EPIC crop growth model [3].

2.10.2.1 Types of Crop Growth Models

Erosion-Productivity Impact Calculator (EPIC) Model. The EPIC model was developed to assess the impact of soil erosion on soil productivity in the United State of America (USA) [33][18]. Because crop yield represents soil productivity, the model is tasked with realistically simulating crop yields for soils with a broad spectrum of erosion damage. Over time, the EPIC model evolved to include application to several agricultural management problems. The model can deal with different management options, such as

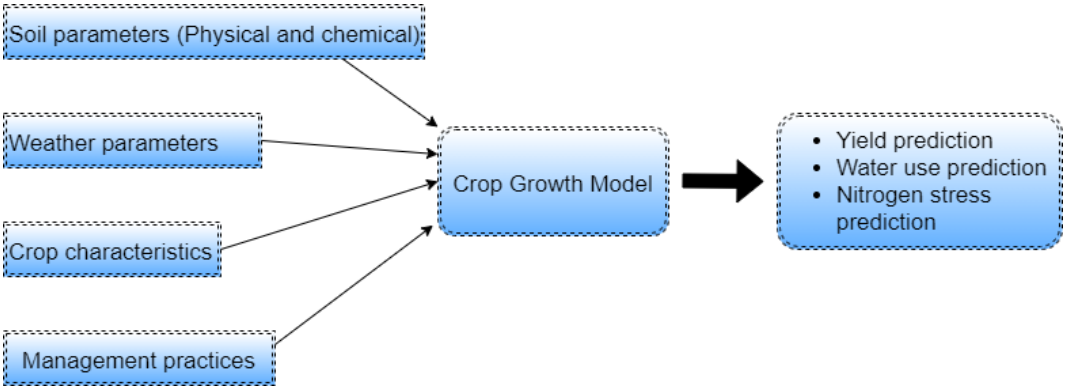


Figure 2.4: Crop Growth Model

tillage operations, planting dates, irrigation scheduling, pest control, and timing and application of fertilizer.
The EPIC model can be divided into nine different components: weather simulation, hydrology, erosion, nutrient cycling, crop growth, tillage, soil temperature, economics, and plant environment control. By considering crop parameters and other relevant factors, the EPIC model can incorporate the calculation of crops' soil water and nutrient intake. It can also predict the impact of factors such as temperature, moisture, nutrients, stresses on crop yield, and examine crop growth progress using a daily step [34].
Since soil productivity translates into crop yield, crop growth is an essential process represented by EPIC. The EPIC model requires a more detailed soil water balance since it was developed to determine soil erosion [35] and includes an additional calculation for taking variation in day length into account .
A drawback of the EPIC model is that it represents all crops with a single crop growth model, but each crop does possess unique values for the parameters of the model.

Decision Support System for Agrotechnology Transfer (DSSAT) Model. The DSSAT is a decision support model that helps to improve operational decisions [57]. The new version of DSSAT, DSSAT-cropping system model (CSM), simulates the growth, progress, and output of a crop growing in a homogeneous area as a function of the soil-plant-atmosphere-management dynamics. Thus, all changes in agricultural components eventually occur under the cropping system and assess the effect of climate variation [57]. However, the DSSAT data requirements are so large that there are few possibilities of usage outside well-equipped experimentation stations, limiting its potential use for land-use planning in developing countries. The CERES-maize is one of the DSSAT cropping model

modules.

Crop-Environment Resource Synthesis (CERES)-Maize Model. The CERES-maize model functions to simulate the growth of maize and its yield. Two different versions of the model exist, including the standard version and the nitrogen version. The former replicates the impact of factors such as weather, genotype, and soil properties on crop growth and yield, and the latter considers the plant nitrogen dynamics and their impact on the crop [10]. In [10], researchers studied the potential of the CERES-maize model to interact with significant plant growth processes in the environment and to respond to weather variation. This was applied over a span of multiple years with data specific to the growing season of a large region. In the CERES-maize model, it is assumed that the model runs in good water conditions. The drawback of this model is that they do not consider the effects of water stress on crop phenology as well as the effects of soil waterlogging on the growth of crops.

Agricultural Production Systems Simulator (APSIM). The APSIM is a modular model that integrates many crop systems [91][92]. APSIM crop simulation models have different modules, such as crop models, soil processes, water balance, fertilizers, erosion and several management controls. It also helps to differentiate models or sub-models on a shared platform. The user can construct a model by determining a group of sub-models from a series of modules related to crop, soil, and utility. A sequence of modules can be defined by the user inputting certain modules and removing any unnecessary modules [92]. APSIM is useful in many applications to aid in farm-related decision-making, management of resources, and seasonal climate forecasting assessments. It can simulate more than 20 crops. APSIM enables users to better understand the effects of weather, soil types, and crop growth management. To predict corn yield, [64] applied the APSIM maize module specifically rendered to the environment of the US Corn Belt.

World Food Studies Simulation Model (WOFOST). WOFOST is a mechanistic model that simulates the growth and output of annual field crops [94][95]. It is also used to simulate the effect of various factors such as climate change on productivity. Moreover, it functions to optimize crop management practices, and the yield of many major crops, including maize, soybean, wheat and potato. The WOFOST model simulates the development of crops from emergence to maturity based on environmental conditions [96]. It can calculate potential yield without considering stress factors, water-limited yield by

considering the water supply, and nutrient-limited yield by considering soil nutrients. In [96], the WOFOST model is used to simulate the growing process of spring maize.

2.10.2.2 Advantages and Limitations of Crop Growth Model

The objective of the crop growth model is to determine whether the model's complexity is suited to a specific problem and whether the model has been tested in diverse environments. [30] presents empirical models as a unique function to evaluate the components involved in the process, but this is insufficient to express the research results. Crop models are limited in their improvement of crop systems' performance and environmental impact assessment. For example, [15] used a model to assess corn yields across tropical, subtropical, and temperate locations in the USA and Australia and concluded that many variations in corn yield were related to temperature variation and solar radiation. They observed that corn yields increased with the increase of solar radiation, and decreased as temperature increased [4]. Table 2.1 summarizes the commonly adopted crop growth models.

2.10.2.3 Yield and Weather Forecasting

Among the industry's concerns is the determination of yield before the harvest. Researchers attempt to improve this area by using historical weather data and employing different techniques. In reality, many weather stations provide data. Still, the challenge is to aggregate and compile up-to-date data and then use a part of it for the remaining seasons. The method is better illustrated by [36] with the CERES-maize model that predicts corn yield by involving the weather data of more than 51 stations in the U.S. In [36], the production was evaluated in three years, and data for the remaining year was used for calibration. Thus, yield forecasting can be estimated by a modelling approach.

2.10.3 Soil Water Availability

2.10.3.1 Total Available Water (TAW)

Soil water availability describes the ability of the soil to store water between two limits: field capacity (the upper limit) and wilting point (the lower limit). After substantial precipitation or irrigation, the soil will deplete until field capacity is reached. Field capacity is the amount of water that remains when descending drainage has significantly decreased. If there is no water supply, the water content in the root zone decreases due to the water

Crop Growth Model	**Characteristics**	**Advantages**	**Disadvantages**
EPIC [33][35][18]	Simulate crop yields for soils with a wide range of erosion damage.	The model has a more detailed soil-water balance.	The model does not provide individual predictions.
DSSAT [57][93]	Incorporates all crops as modules using a single soil model.	Run in well-equipped experimentation stations.	Large data requirements
CERES [10][30]	Two different versions: standard and nitrogen.	The model assumes that it runs in good water conditions.	Does not consider the effects of water stress on crop phenology and soil water-logging on crop growth.
APSIM [91][92]	Shares the same modules for the simulation of the crop components.	Users are able to integrate a model by determining a set of submodels from modules.	Unable to simulate complete nitrogen dynamics in complex farming systems.
WOFOST [94][95][96]	Unable to simulate complete nitrogen dynamics in complex farming systems.	The model can dynamically describe the fundamental processes of crop growth, such as photosynthesis, respiration, transpiration and biomass partitioning.	Does not simulate processes of soil N dynamics, Some parameters are fixed whereas in practice they are known to vary.

Table 2.1: Summary of commonly adopted crop growth models

uptake of the plants. As the uptake of water advances, residual water is retained in the particles of soil, reducing their energy potential and rendering it difficult for the plant to extract [53].

At some point, the remaining water can no longer be removed by the plants. Hence, the uptake of water becomes nonexistent as the plant reaches its wilting point. The wilting point represents the soil water content level that results in the permanent withering of a plant [53][98].

2.10.3.2 Readily Available Water (RAW)

The uptake of water by the crop decreases considerably before it reaches wilting point, despite its theoretical availability before that point. When the soil is moist enough, the soil supplies sufficient water to match the crop's atmospheric demand, and the uptake of water is equal to the evapotranspiration. Water then renders itself wholly bound to the soil matrix when the soil water content evaporates, and it becomes more challenging to extract. Beneath a specific threshold value, water in the soil cannot be sufficiently transported to the roots to address transpiration demands [54] [98].

2.10.4 Growth Constraints

Generally, crops do not reach their potential growth and yield due to a variety of environmental limitations. The model evaluates the risks of the stresses generated by several factors such as water and nutrient content, temperature, aeration, and radiation. Predicted stress factors vary between 0.0 to 1.0, and can impact plants in several ways.

2.10.4.1 Biomass

The biomass constraint is estimated using the least value amongst the estimated stress factors for nutrients and water. If one of the plant stress factors is below 1.0, the daily biomass accumulation is adjusted as the following equation shows:

$$B = B_{p(i)}(REG) \tag{2.1}$$

REG is a factor that regulates the growth of crops (the least value from among the estimated stress factors).

2.10.4.2 Water Stress

Water stress is primarily responsible for significantly impacting the plant's phonological aspect, especially photosynthesis [19]. The final grain number is determined between two weeks before silking and two to three weeks after silking. During that crucial period, the crop needs a constant water supply; a longer duration of water stress will cause the crop to fail.
Water stress decreases grain yield due to its significant reduction of the total leaf area, leaf expansion, and biomass, and its increase of senescence rates [20]. The harvest index might be reduced by water stress during the critical crop stages, usually between 30 and 90% of maturity.

The water stress factor is estimated by considering supply and demand. [53] evaluates water stress by considering the total amount of water that a crop can extract from its root zone, and its estimation depends on the type of soil and root depth. Further, it considers the RAW in the root zone and the water deficit relative to field capacity (root zone depletion). Root zone depletion is zero at the field capacity.

According to [21], drought stress is a factor that limits biomass production in proportion to transpiration reduction.

2.10.5 Precision Agriculture End-to-End Platform

PA end-to-end platforms provide farmers with different sets of features and capabilities throughout the growth cycle to maximize yield, improve equipment and labour, reduce costs and produce high-quality crops. The platforms consist of hardware and software and integrate with external satellite data or other sources.

In many cases, the platforms operate with other data management information systems to generate a comprehensive view of the farming process and communicate the most timely, relevant analytics, reports, and recommendations possible. There are several platforms in precision agriculture, such as Farmers Edge, Trimble, and FarmBeats. In this work, we will be focusing more on the design of FarmBeats.

2.10.5.1 Farmers Edge

Farmers Edge is a platform dedicated to developing solutions for smart farming. Farmers Edge comprises farm-related hardware, user-friendly software, digital agronomy, and AI-

based analytics support. It provides farmers with everything they require to benefit from farm data, manage risks, and maximize production sustainably and at an affordable price.

Data Collection is performed by gathering data from the farm's weather stations and telematics devices. There are various data sources in the Farmers Edge, such as satellite imagery, on-farm weather stations, soil moisture probes and CanPlug.

The high resolution of satellite imagery can convert imagery into timely, field-level insights in order to distinguish problems as they emerge. In addition, they have a fast data processing engine that assures images are dispatched within 48 hours of conception. The images are processed into readable field maps to monitor the health of crops and any changes in crops and soil.

The weather stations serve to track essential factors, including temperature, wind speed, wind direction, dewpoint and rainfall. They also have access to several other weather stations. In addition, they provide better data, power many models and provide severe weather alerts.

Soil moisture probes are water-centered tools that support decisions and measure real-time moisture. They also function to observe the water content of root zones using layered soil moisture and temperature measurements at six depths. In addition, they enhance reporting and yield prediction.

CanPlug are in-field telematics and data transfer devices that help to transfer data. Data transfer is secured and stored automatically. They also provide real-time equipment monitoring.

2.10.5.2 Trimble

Trimble is a farm management platform that enables farmers to accurately monitor and map field information in real-time. It also assists in achieving functional changes to maximize profitability through the entire farm operation.

Trimble uses sensors, and primarily UAV technologies, to collect data from fields. UAV technologies are one of the most inexpensive and easy-to-operate agricultural devices. They have enabled a significant reduction of working hours and increased stability, measurement accuracy, and productivity. Additionally, Trimble includes features such as weather forecast and creates client, farm and yield records. It features different services that can enhance farm operations, including:

Trimble connected farm - which comprises software, hardware, and correction service; helps simplify workflows, and increases efficiency and profitability in farm operations. In addition, it allows farmers to manage and organize crops, perform farm operations, follow all occurrences in the field, and have a substantial history of the total crop yield.

Trimble's WeedSeeker 2 is a spot spray system that allows the detection and elimination of resistant weeds. The system uses advanced optics and processing power.

Trimble FieldLevel II system helps farmers eliminate high and low spots in their field to maintain water distribution and improve overall crop yields.

2.10.5.3 FarmBeats

In [9], the author presents an end-to-end IoT platform for precision agriculture called FarmBeats that allows the gathering of data from different sensors such as cameras, drones and soil sensors. Each type of sensor differs in terms of bandwidth constraints. FarmBeats performs with infrequent deployment of sensors to ensure the system's availability even during power or internet outages due to inclement weather - scenarios that are relatively common for a farm. FarmBeats allows connectivity with the cloud to store data permanently and perform data analytics.
The main challenges that FarmBeats is designed to solve are described below.

1. FarmBeats works with two different data streams, including visual data from UAVs and sparse sensor data from ground sensors. FarmBeats combines data from several streams and enables the deployment of sensors infrequently. As noted in [9], UAVs' measurements are dense in space but sparse in time, unlike the ground sensors, which are sparse in space but dense in time. FarmBeats set up a specific graphical model in order to integrate different streams to produce dense space-time sensor maps of the farm. The FarmBeats system employs the teacher-student model in order to train the graphical model to correspond between visual features and sensor values. The sparse sensor deployments provide the training data.

2. Another challenge FarmBeats works to solve is the installation of a reliable high bandwidth link between the farmer's home internet and the IoT base station connection. FarmBeat's solution to this issue of ensuring a better connection within the farm was influenced

by the latest research on unlicensed TV white space (TVWS). Since the farm often faces power or internet outages, the IoT base station must be powered by battery-backed solar power, which is limited by weather conditions. Resolving this problem will require a new design of weather-aware IoT power scheduling, which uses weather forecasts to duty cycle the base station components (initialize base station, connect to sensors, read from sensors).

3. Internet connection to the cloud is typically slow on a farm; therefore, sending high bandwidth drone videos to the cloud is infeasible. The answer to this challenge is to design a fog-based solution that will allow the computational process to be performed locally on the farmer's PC and support low latency applications and services. The data will be aggregated and transmitted to the cloud for long-term analytics. The FarmBeats gateway also allows facing potential internet outages [9].

4. Drones are well-known sensors that farms use to get aerial images but have limited battery life and take time to be recharged; plus, farms are open space, and flying against the wind makes drones consume more battery. An innovative algorithm has been designed in the FarmBeats gateway to solve this problem. The solution consists of leveraging wind and enabling the drones to accelerate and decelerate in order to save battery life and optimize flight paths. This algorithm is driven by how sailors use winds to navigate sailboats.

5. Deploying sensors all over a farm might be costly and can also interfere with a farmer's daily activities. The solution to this challenge is to establish a mechanism that combines both drones' aerial imagery data with sparse ground sensors to produce precision maps for the farm.

2.10.5.4 FarmBeats Architecture

Sensors and Drones: Sensors deployed in the field measure the physical parameters, including soil moisture and soil pH and transmit data to an IoT base station using an internet connection such as Wi-Fi. FarmBeats also uses cameras and drones to take aerial imagery and regularly send them through a specific communication network. An app is installed on the farmer's phone to schedule the UAV flight either periodically or manually.

IoT Base Station: The IoT base station is powered by battery-backed solar power and comprises the TVWS device, the sensor connectivity module and the base station

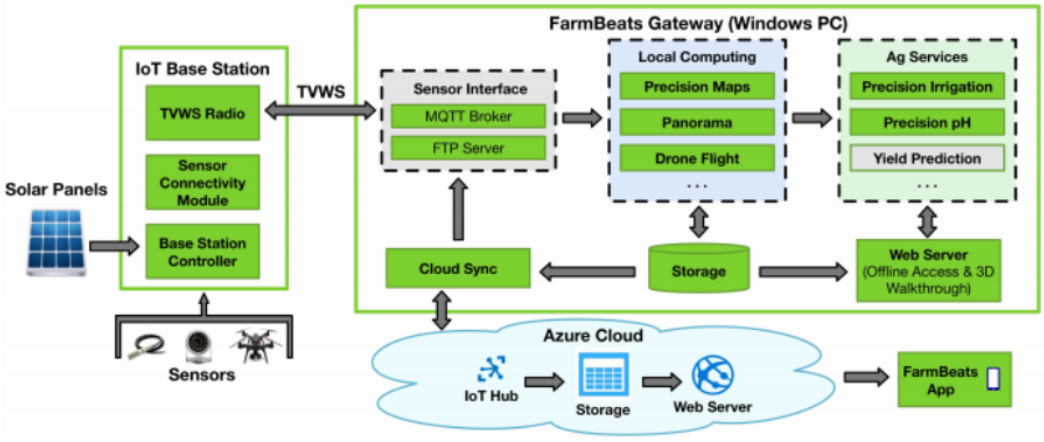


Figure 2.5: FarmBeats Architecture [9]

controller.
- The TVWS device collects and stores data in the base station on the farm and ensures that it is transmitted to the gateway and then to the cloud for long-term analytics.
- The sensor connectivity module connects the base station with the field sensors. In the FarmBeats design, the module is simply a Wi-Fi router.
- The base station controller plays two different roles::
1. It is utilized as a cache for the sensor data gathered by the sensor module. While the TVWS device is turned on, the cache syncs the data with the IoT gateway.
2. It schedules and implements the duty cycle rates by considering the present battery status and weather conditions.

IoT Gateway: The IoT gateway performs data aggregation from stored data and transmits it to the cloud. The farmer uses a PC as FarmBeats gateway, located in his office with internet access; the IoT gateway allows the farmer to post and access web services when connected to the farm network. Without any connectivity to the cloud, FarmBeats can still be available and have local access to the data. Before the aerial drone data can be sent to the cloud, the IoT gateway runs built-in algorithms that plan the drone path and reduce the data.
FarmBeats differs from the prior IoT gateway in terms of web service design. The service implemented into the FarmBeats gateway is different from the one in the cloud. The

FarmBeats gateway can work without internet connection and continue providing essential services.

Services & Cloud: The IoT gateway sends the aggregate data to the cloud. Then, the latter permanently stores the data for further analysis and decision-making and provides the farmer with the ability to access data through a web interface. FarmBeats uses the Gaussian process model for field measurement prediction. Machine learning models similar to the Gaussian process may be applied to provide better results in different farming fields.

2.11 Machine Learning

Machine learning is a type of artificial intelligence that equips systems with the power to learn from experiences and enhance their performance without being specifically programmed [27].

Learning from past data, ML builds models to predict the outcome with a higher accuracy level. A higher amount of quality data leads to increased accuracy of the models.

Generally, machine learning algorithms are categorized into four groups: supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning.

2.11.1 Supervised Machine Learning

Supervised learning refers to a machine learning technique that learns from labelled training data, which has inputs and desired outputs. Supervised learning is conducted based on ground truth, which indicates that previous knowledge of output values will already be known.

Supervised learning is performed using two different approaches: Classification is when the input data and the discrete output values are mapped. Regression is when the input data is mapped to continuous output values. Several supervised learning techniques such as nearest neighbor, naive Bayes, decision trees, support vector machines (SVM) and neural networks exist.

An algorithm's choice is based on its characteristics such as the speed of training, memory usage, flexibility, prediction accuracy, and how easily results can be interpreted.

2.11.1.1 K-Nearest Neighbor (KNN)

KNN is a supervised machine learning technique based on a classification method where there is no assumption about the distribution of the dataset and the input is classified based on the similarities between neighbours. The similarity between neighbours is based on a particular measure of distance such as Euclidean or Manhattan distances. This distance is defined as follows:

$$d(E_i, E_j) = \sqrt{a \sum_{i=1}^{n} (E_{i,a} - E_{j,a})^2} \tag{2.2}$$

Where d is the distance between training and testing data, E_i and E_j are n-dimensional vectors and a is the data sample.

The KNN classifier computes the distance between E_i and E_j in the dataset and each training observation. The K points that are closest to E_i form a set a. K is usually odd to prevent tie situations. The advantage of the KNN technique is that all the descriptive features are considered in the prediction. It can handle the noise and the result is easy to interpret. The drawback is that the KNN performs slowly with a large dataset, and the upper bound on accuracy is generally lower [28].

2.11.1.2 Naive Bayes

Naive Bayes is a supervised machine learning technique that uses Bayes' theorem with an assumption of independence of the input feature. Naive Bayes is a classification technique that assumes that the presence of features in a particular class are not related to any other.

Bayes' theorem helps to calculate the posterior probability distribution of the output vector y or a target given the input vector of features as follows:

$$P(y|E) = p(E|y)p(y)p(E) \tag{2.3}$$

Where $P(y|E)$ denotes the posterior probability of the output y regarding the input vector of the feature. $p(E)$ is the prior probability of the input vector of the feature E, $p(y)$ denotes the input vector's prior probability, and $p(E/y)$ denotes the likelihood probability distribution of the feature's input vector.

The advantages of naive Bayes models are that they perform well with a small dataset and are easy to interpret. The drawbacks are that they struggle with a continuous target and require large storage.

2.11.1.3 Decision Tree

Decision tree is a supervised machine learning technique that classifies the data using a tree-like model of decisions and possible outputs. A decision tree is an algorithm that covers both classification and regression problems. There are three different types of nodes in the decision tree classifier: root node, internal node and leaf node.

The decision node is the feature of the dataset to be classified, and the branches are the values that the node can predict. To determine the information gain, we first define the entropy that specifies and describes an arbitrary collection's purity. Generally, decision tree algorithms are well known as classification and regression trees (CART).

The advantages of CART are that they are simple to understand, interpret and visualize, perform quickly with a large dataset, and the non-linear relationships between parameters do not influence the tree performance. The disadvantage of CART is that it is not ideal for continuous data. Only a subject of features is checked when traversing the tree instead of considering all features. This is called overfitting.

2.11.1.4 Random Forest

Random forest (RF) is the most accurate classifier machine learning technique. Built on the decision tree algorithm, the random forest uses a set of trees for the classification. Enriched ensembles are developed by constructing random vectors, which help generate each tree from the random vector.

Random forest uses the output of trees to solve the classification and regression problem. There is a difference between the random forest problem for classification and regression. Random forest for classification receives a class vote from each tree and then categorizes using a majority vote among the predictions. When the random forest is used for regression, the predictions from each tree at a target point x are averaged.
The vote made by the tree helps to predict a decision and improve stability and accuracy, and the majority defines the final random forest prediction.

Random forest tends to overfit the data when the number of variables is large, but the fraction of relevant variables is small. At each split, the probability can be so slight that the relevant variables will be selected, and the algorithm may potentially perform inaccurately. Random trees correct the decision trees' tendency to overfit training sample data.

Algorithm 2.1 Random forest algorithm pseudo code

To generate c classifiers:

for $i = 1$ to c **do**

Randomly sample the training data D with replacement to produce D_i

Create a root node, N_i containing D_i

Call BuildTree(N_i)

end forBuildTree(N):

if N contains instances of just one class **then**

return

else

Randomly select x% of potential splitting features in N

Select the feature F with the highest information gain to split on

Create f child nodes of N, $N_1, ..., N_f$, where F has f potential values $(F_1, ..., F_f)$

for $i = 1$ to f **do**

Set contents of N_i to D_i, where D_i is all instances in N that match F_i

Call BuildTree(N_i)

end for

end if

2.11.1.5 Support Vector Machine (SVM)

SVM is a supervised method of ML that tries to find a hyperplane to split the data and maximize the margin of the classifiers. Choosing a kernel function is essential, which decreases the amount of support vectors. Some frequently utilized kernel functions include linear, polynomial, and Gaussian radial basis functions. The linear kernels can be defined as:

$$K(E) = w^T E + b \tag{2.4}$$

Where E denotes the energy vector corresponding to all the frequency channels. w denotes the weighting vector, $(.)^T$ denotes the transpose operator, and b is the bias.

The linear support machine is an optimization problem over the weight and can be illustrated as shown:

$$min_{w \in R^d} ||w||^2 + C \sum_{i}^{N} max(0, 1 - y_i K(E_i)) \tag{2.5}$$

Where y_i is the i_{th} element of the output, C is a regularization constant, and E_i is the energy statistic of the i^{th} element in the training dataset. The polynomial kernels can be quadratic (degree 2) or cubic (degree 3) and are defined as:

$$K(E, y) = (E.y + 1)^d \tag{2.6}$$

Where E E signifies the energy vector of the received samples of all frequency channels going through the sensing process, y is the output vector whose elements take two values 0 and 1 and d is the polynomial degree. Gaussian radial basis function kernel can be expressed as:

$$K(E, y) = e^{(\gamma - ||E - y||)^2} \tag{2.7}$$

Where E denotes the energy vector corresponding to all the frequency channels, and y is the output vector and is a constant.

2.11.1.6 Neural Networks

Neural networks, otherwise referred to as artificial neural networks (ANNs), are complex models that mimic the operation of the human brain. The ANN comprises several connected neurons, like the biological neural system, which analyzes and processes information. Neural networks are multi-layer networks, including the input, hidden, and output layers, as shown in Figure 2.6.

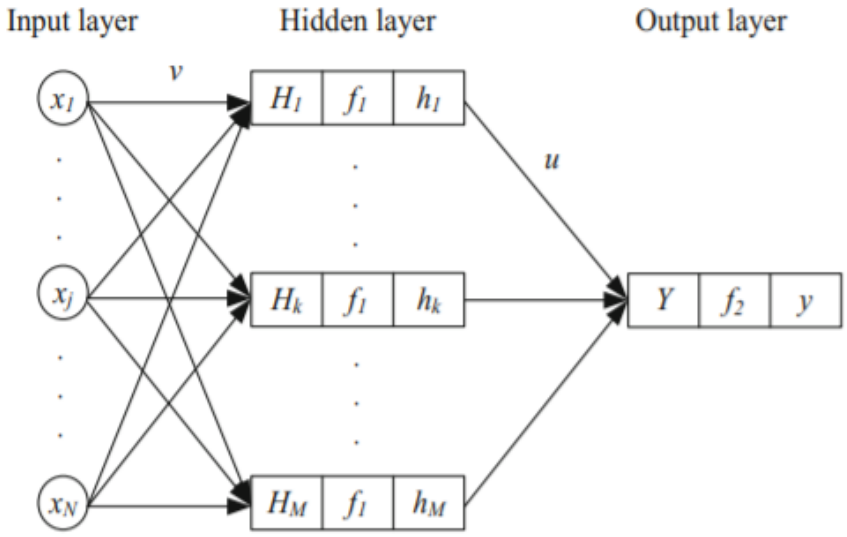


Figure 2.6: Artificial neural network model for crop yield [102]

An artificial neuron is connected to others and has an associated weight and threshold. It executes three basic operations: addition, multiplication, and activation. A specific neuron is activated when the output is above the threshold value and can transfer information to the network's next layer. Neural networks depend on the training of data to understand and improve their accuracy over time. The output layer of a neuron is expressed as follows:

$$Y = \sum_{j=1}^{M} u_j h_j + \beta \tag{2.8}$$

Where h_j is propagated forward to the output layer, u_j indicates weight and β denotes the bias value added.

Multilayer Perceptron (MLP) is viewed as one of the neural networks in which weights and biases can be trained to reach a specific target. The MLP is trained using the backpropagation of error to modify the weights and decrease errors.

The perceptron is made up of fully connected input and output layers and may contain multiple hidden layers in-between, as shown in Fig. 2.6
The MLP algorithm is as shown [112]:

1. The inputs are advanced forward through the MLP by combining the scalar product of the input along with the weights found between the input layer and the hidden layer. The scalar product then generates a value at the hidden layer. This value is not pushed forward.

2. Activation functions are used by MLPS at corresponding, calculated layers. Many activation functions are utilized, such as linear, rectified linear units (ReLU), sigmoid function, and tanh. The output that is calculated is pushed at the present layer using any of the activation functions.

3. Upon the hidden layer's calculated output being pushed through the activation function, it is moved to the subsequent layer in the MLP by taking the scalar product with its matching weights.

4. The second and third steps are repeated until reaching of the output layer.

5. When the output layer is reached, either the calculations will be utilized for a back-propagation algorithm corresponding to the selected activation function for the MLP (if training), or a decision is reached using the output (if testing).

2.11.2 Unsupervised Machine Learning

Unsupervised learning is method of ML that uses algorithms to independently learn and generate decisions on unlabeled data. The algorithm works on its own to find structure from input data. Contrary to supervised learning, an unsupervised learning algorithm can perform more complex tasks.

The objective of an unsupervised algorithm is to discover unknown patterns in data and identify and categorize features. Unsupervised learning is commonly used to identify structure in data and apply dimensionality reduction. The unsupervised algorithm derives mining rules from inputted data, discovering hidden patterns, and categorizing and describing data to derive meaningful insight.

2.11.2.1 K-means Clustering

K-means clustering is a popular unsupervised ML technique that groups data into clusters; in other words, it helps find a pattern in a collection of uncategorized data and places them into a specific group. In the clustering approach, the quantity of clusters is selected a priori

and data is categorized into k groups called centroids. Data that have similar properties will belong to the same group.

The primary goal is to specify k centroids, with one centroid for every cluster. The following step is to assign each data point to the closest corresponding centroid, using a particular distance measure such as Euclidean distance. Once the allocation is done, the k centroids are recalculated. The process continuously repeats until no changes occur in the centroid values, indicating that they have been accurately grouped.

2.11.2.2 Apriori Algorithm

The Apriori algorithm is an unsupervised algorithm that enables the creation of associations between data. The technique is called association rules, and it helps to discover relationships between data in large databases. Association rules use the if-then format to find relationships in the data, which means that if a group of particular events occurs frequently, then all of the group's subsets will occur frequently as well.

The parameter used in this technique is the minimum support; it helps extract data to determine associations in the dataset. It is also used to define how frequently particular events occur in the dataset. The association rule is the procedure of discovering hidden patterns in a large dataset.

2.11.2.3 Markov Chain

Markov chain is a statistical Markov model based on the probability of choosing the best state among random variables. In this method, we assume that the system which is modelled follows a process with hidden states. Markov chain helps to compute the probability distribution to forecast future conditions using likelihoods based on current and past states. In the Markov chain method, the probability with the greatest result will dictate the future state.

2.11.2.4 Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

DBSCAN is an unsupervised learning method that clusters data together based on the similarity of observations. The DBSCAN algorithm basically requires two parameters, epsilon (eps) and the minimum points (MinPts). Epsilon designates the proximity that

Parameters	**Supervised machine learning method**	**Unsupervised machine learning method**
Input Data	Algorithms are trained using labelled data	Algorithms are applied against unlabelled data
Computational Complexity	Simpler method	Computationally complex
Accuracy	High level of accuracy and reliability	Less accuracy and reliability
Real-Time Learning	Learning occurs offline	Learning occurs in real-time
Quantity of Classes	The quantity of classes is known	The quantity of classes is unknown

Table 2.2: Supervised vs. Unsupervised Machine Learning

points should have in order to be deemed as part of a cluster. What this means is that if the measured distance between two points is less than or equal to the epsilon value, the two points are regarded as neighbours. Moreover, the minimum points specify the least number of points required to create a dense region.
For instance, if we set the MinPts parameter as one, at least two points are needed to form a dense region.

2.11.3 Reinforcement Learning

Reinforcement learning is a machine learning technique that allows an intelligent agent to learn in an interactive environment by trial and error and perform actions to gain rewards [29]. Similar to supervised learning, reinforcement learning also has input data and desired output. The difference is that reinforcement learning uses the notion of cumulative rewards and punishment as the action for a precise behaviour, whereas unsupervised learning consists of discovering hidden patterns in unlabelled datasets.

The reinforcement learning procedure entails learning what action to take in a specific situation to accumulate rewards. Since the activities of the learning system have an influence on its later inputs, they are effectively considered closed-loop problems.

Contrary to other machine learning algorithms, the learner does not know the action to perform prior, but must instead learn through trial and error and perform actions to acquire more rewards.

In some cases, the performed actions have a direct impact on the next situation and the

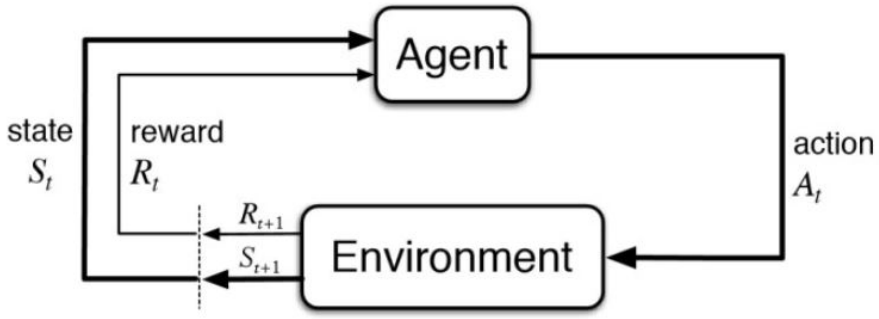


Figure 2.7: Reinforcement Learning Model

rewards. Hence, there are three critical features in reinforcement learning- its closed-loop nature, the uncertain environment due to lack of prior knowledge, and its ability to play out over extended time periods [28].

2.11.4 Machine Learning in Precision Agriculture

IoT application in agriculture is based on different functions such as monitoring, precision agriculture, and greenhouse production. Data acquired in crop farming help farmers obtain greater insight of the environmental factors affecting their productivity. Humidity, air temperature, soil temperature, solar radiation, and wind speed are some of the weather elements monitored in smart farms [27].

In [6], an intelligent system is presented that can predict soil moisture based on information collected from the sensors deployed on the field and the weather forecast information available on the Internet. An algorithm based on hybrid machine learning methods has been developed for soil moisture prediction. The algorithm is a combination of supervised and unsupervised machine learning techniques.

The support vector regression (SVR)and K-means clustering methods are combined to provide high accuracy. K-means clustering estimates variations in soil moisture based on weather conditions. The result shows improved accuracy and less mean squared error (MSE).

2.11.5 Machine Learning in Smart Farming

Significant research has been conducted to leverage machine learning in agriculture to acquire and assess data, achieve more precise predictions, and manage crop components [27]. ML has helped to predict crop yield, identify crop disease, manage critical components (water, nutrients), and much more.

In [2], an ML-based framework is proposed to predict corn yields in the three US Corn Belt states of Illinois, Indiana, and Iowa. Additionally, information on complete and partial in-season weather is taken into consideration. Results showed that individual models outperformed ensemble models established upon a weighted average of base learners (average ensemble, exponentially weighted average ensemble, and optimized weighted ensemble). Compared to other developed models, the proposed ensemble model could individually accomplish the greatest accuracy of prediction and the least mean bias error. Therefore, they designed multiple ML and ML ensemble models with a sequential blocking method to produce out-of-bag estimates and assess their performance in corn yield predictions [58]. They also investigated the impact of accounting for complete or partial in-season weather data when predicting outcomes.

In [64], researchers designed various ensemble models using a blocked sequential procedure to produce out-of-bag estimates. Also, a weighted ensemble model that was optimized was proposed. This model accounted for predication variances and biases and incorporated out-of-bag estimates to determine the ideal weight for combining more than one base learner. Ensemble models that perform well necessitate that base learners demonstrate a specific level of "diversity" in their predictions and individually retain high performance. Therefore, an assortment of models were chosen and trained, such as linear regression, LASSO regression, extreme gradient boosting (XGBoost), LightGBM, and random forest. Also, several two-level stacking ensemble models, an average ensemble, and an EWA, were created and assessed on unseen test observations. Observations concluded that machine learning ensembles outperform a single machine learning technique, are capable of reducing variance, prediction bias or both, and can more readily determine underlying data distribution.

[8] compared several AI models to produce the greatest prediction model for crop yield for the Midwestern US. This included models such as random forest, support vector machine, neural network, artificial neural network and deep neural network. After a high-resolution CDL, the inputted dataset was created from meteorological and hydrological variables, and vegetation indices based on satellite. They then used the database with optimal inputs to construct six key AI models for crop yield prediction and comprehensively and objectively compared the AI models. For the mean absolute error (MAE), DNN

outperformed the five other AI models.

In [9], FarmBeats, an end-to-end IoT platform for data-driven agriculture, provides a platform for collecting data from various sensor types, i.e., cameras, drones, and soil sensors. FarmBeats can resist a power outage and may include different services.
Significant features of FarmBeats include long-term and large-scale deployment. A novel weather-aware IoT base station is designed to reduce the base station downtime to zero. Novel inference techniques and a wind-assisted drone flight planning algorithm are designed to compress the aerial imagery data. In addition, FarmBeats uses television white spaces to connect IoT base stations to the Internet, effectively solving bandwidth issues. The observation is that FarmBeats uses a Gaussian process model.

In [58], the prediction of crop yield production was performed by using time series data. First, they proposed SVM and naive Bayes as classification methods. Then, they compared them with proposed ensemble Adaboost methods, such as AdaSVM and AdaNaive, to predict crop production over time. The analysis results showed that the two ensemble methods provided better prediction accuracy and reduced the classification error more than the classification methods. Furthermore, the results of the proposed methods may differ if the dataset input increases for the same techniques or if different methods are employed.

Reference	Types of Application	ML Models	Description
[1]	Cluster analysis	K-Means and Hybrid K-Means clustering	Hybrid K-means has better performance than K-means in terms of time complexity and accuracy
[2]	Yield forecasting	Linear regression, Lasso, XGBoost, Random forest, Weighted ensembles, Stacked ensembles	Weighted ensembles outperform stacked ensembles and individual models by providing more precision and less errors
[8]	Yield prediction	Random forest, SVM, ANN, DNN	DNN outperformed the other AI models in terms of prediction errors
[58]	Yield prediction	SVM, Naive Bayes, Ensemble models (AdaSVM and AdaNaive)	Ensemble models provide better prediction accuracy and reduce classification errors
[6]	Irrigation management	SVR, SVR + K-means	SVR + K-means model has higher accuracy and lower errors than the SVM model
[85]	Yield estimation, Climate anomalies	Gaussian process (GP), Linear regression	GP models outperform ML models in terms of accuracy, robustness and errors of the predictions
[86]	Yield estimation	Deep Learning and SVM	Deep learning with two InnerProductLayer provides high accuracy and outperforms the SVM model
[87]	Corn Yield estimation	SVM, RF, Extremely Randomized Trees (ERT) and Deep Learning	Deep learning provides the highest accuracy and overcomes the overfitting problem
[88]	Soil properties and Corn Yield prediction	SVM with radial and linear kernel functions, RF, Neural Network (NN), Gradient Boosting Model (GBM) and Cubist (CU)	NN model provided the most accurate prediction for soil organic matter (SOM). CU produced the most accurate prediction for the cation exchange capacity (CEC). The RF model produced the most accuracy for the corn yield.
[89]	Corn yield prediction	Neural Network techniques	The corn yield was best predicted using Back-propagation Neural Network (BPNN)
[90]	Irrigation management	Classification and Regression Trees (CART), Random forest (RF), and conditional inference trees (Ctree)	The corn yield was best predicted by using Ctree
[97]	Corn yield prediction	Artificial Neural Network (ANN), Multiple Linear Regression (MLR)	The results showed that ANN prediction is more accurate than MLR-based yield model

Table 2.3: Machine Learning in Smart Farming

Chapter 3

Yield Prediction

This chapter presents mechanistic crop growth and machine learning models and shows how the combination of the two assists in predicting and optimizing crop growth and yield.

3.1 Mechanistic Crop Growth Model

In this section, we present a mechanistic corn growth model and how to optimize it to consider factors that critically impact potential corn growth and yield, such as water and nutrients. We also explain its application in managing risks and growth constraints to achieve better results compared to previous works.

3.1.1 Maize Growth Model

Maize is one of the most significant crops in an agricultural environment. Maize cultivation is a worldwide endeavour and helps solve food security issues. It can be used for food, livestock feed, fuel and industrial products, and is produced annually. Studies show that the average maize yield has critically increased due to appropriate crop model developments in recent years [52].

The mechanistic crop growth model is a mathematical description of all physiological processes throughout the growing season. It entails a dynamic simulation process of an entire crop by predicting the growth of its components such as leaves, roots, stems and grains and considers field farming interrelationships. Therefore, the aim of crop growth

simulation models is to forecast the final condition of biomass or harvest-worthy yield and includes quantitative data about important processes that occur in the growth and development of plants [10].
Yearly crops grow from date of planting to harvest time or until total heat units are equal to the potential heat units of the crops.

$$HU_i = \frac{(T_{max,i} - T_{min,i})}{2} - T_b, j \tag{3.1}$$

HU_i, $T_{max,i}$, and $T_{min,i}$ represent heat unit values, and maximum temperature, and minimum temperature (°C) on the day i, and $T_{b,j}$ is the base temperature specific to the crop in question, which is j (growth does not occur at or below T_b). The base temperature most commonly used for all phenological phases is 8° C, except for silking (seedling emergence) [10]. The heat unit is rendered zero if the maximum temperature is less than the base temperature.
The crop's phenological progression is dependent on daily heat unit accumulation. The following equation represents the relevant computation.

$$AHU_{(i+1)} = AHU_i + HU_i \tag{3.2}$$

$AHU_{(i+1)}$ is the accumulated heat units on day $i + 1$.

The expansion of the leaf is an essential factor determining the competitive ability of the crop. The leaf number is used as a measure of development rates and is influenced by the soil temperature and photoperiod. When the plant emerges and the number of leaves is fewer than the maximum, the leaf number is calculated as the exponential function of accumulated heat units with the base temperature [11].

$$LN_{(i)} = 2.5 * e^{(AHU_{(i)} * 0.00255)} \tag{3.3}$$

[12] developed an equation that calculates the leaf number (LNM) with the largest area from the total number of leaves initiated (TLN).

$$LNM = 3.53 + 0.46 * TLN \tag{3.4}$$

The value of TLN must be initially defined.

Several methods have been suggested to measure the leaf area due to its importance for crop light interception and its large influence on crop growth and final yield [13]. It has

been proven that the area of an individual leaf can be surmised from leaf number (LN) and the area of the largest leaf (AM) [5].

$$A_{(i)} = AM * e^{[-0.0344*(LN_i - LNM)^2 + 0.000731*(LN_i - LNM)^3]} \tag{3.5}$$

The observation made in [13] concludes that if the area of all expanding leaves on a plant at a given time are combined, they are equal to the completely expanded area of the two developing leaves right above the last leaf that fully expanded. As a result, before the last two leaves on a plant fully expand, the calculation of the total leaf area per plant is presented as the total sum of all fully expanded leaves and the fully expanded leaf area of the subsequent two leaves [14].

$$\text{Plant Leaf Area} = \sum_{n=1}^{LN+2} A \tag{3.6}$$

[13] suggests an exponential relationship that computes the fraction of the total leaf, which was senesced (FAS), increased with heat units (HU) from emergence.

$$FAS_i = 0.00161 * e^{0.00328*(AHU_i - 87)} \tag{3.7}$$

In the majority of crops, the leaf area index (LAI)) starts at zero or close to that. It then exponentially increases in the stage of early vegetative growth. During this time, the rates of leaf primordia development, leaf tip appearance, and blade expansion are linear functions of heat unit accumulation. LAI is calculated as the difference between total (PLA) and senesced leaf area per plant (FAS) multiplied by the population of plants.

$$LAI_{(i)} = \begin{cases} \text{Plant Population} * (PLA - PLA * FAS_{(i)}) & \text{if LAI greater than 0} \\ 0 & \text{Otherwise} \end{cases}$$

3.1.2 Potential Growth

Using Beer's law equation, energy interception is approximated as a function of solar radiation and the LAI of the crop.

$$PHS_i = 0.5 * RA_i[1 - e^{(-0.4*LAI_i)}] \tag{3.8}$$

PHS_i is the radiation use efficiency obtained from photosynthesis, RA is solar radiation, 0.5 is the ratio between PHS_i and RA_i (the radiation use efficiency), and LAI is leaf area

index.
The possible daily increase in biomass can be approximated as the product of intercepted energy and a crop parameter for transforming energy into biomass [15].

$$BM_{(i+1)} = (BM_i) + (PHS_i) \tag{3.9}$$

BM_i is the daily potential accumulation in biomass, $BM_{(i+1)}$ is the daily potential increase in biomass on day $i + 1$.
Harvest index increases as a nonlinear function of heat units. It varies from zero at the planting stage to the optimal value at maturity. At the end of grain filling, i.e., 33 days after the planting stage, the harvest index equals 0.5.

$$HI_{(i+1)} = \begin{cases} 0 & \text{At the planting stage} \\ HI_i + 0.015 & \text{Otherwise} \\ 0.5 & \text{33 days after planting stage} \end{cases}$$

The crop yield is estimated through an integrated calculation of the potential biomass and harvest index concept.

$$GRAIN_i = HI_i * BM_i \tag{3.10}$$

Where $GRAIN_i$ is crop yield, HI_i is harvest index, and BM_i is the daily potential accumulation in biomass

3.1.3 Water Availability and Nitrogen

3.1.3.1 Water Availability

Soil Water Balance The efficient use of water has a massive impact on crop yield and is considered the most critical agricultural factor. Water excess and shortage can equally have a significant effect on the quality and quantity of yield. Water directly affects photosynthesis, respiration, absorption, translocation and utilization of nutrients, among many other processes. It similarly affects evaporation and water transpiration through the plant stems and tissues [16]. Whenever there is an excess of water, the roots can rot, and oxygen cannot be supplied to the crops. Conversely, when there is a shortage of water, vital nutrients cannot travel to the plants [53] [98].

Soil water inputs could be from rainfall and irrigation and be removed by soil evaporation and crop transpiration.

There are two different layers for soil water: the top layer of soil water ($SW1$) and the entire soil water layer (SW).

$$SW1_{(i)} = \begin{cases} SW1_{(i-1)} + RF_{(i)} + Irr_{(i)} & SW1 < MSW1 \\ 19.5 & \text{Otherwise} \end{cases}$$

$$SW_{(i)} = \begin{cases} SW_{(i-1)} + RF_{(i)} + Irr_{(i)} & SW < MSW \\ 135 & \text{Otherwise} \end{cases}$$

Where $SW1$ and SW are the top layer and the entire soil water respectively, $MSW1$ is the maximum (total) soil water in the top layer and MSW is the maximum soil water in the entire soil water. RF is the rainfall and Irr is the irrigation.

The soil water is lost due to soil evaporation SEP and crop transpiration TR. There are two stages for assessment of soil evaporation. In stage I, the soil evaporation (SEP) is evaluated using the FAO Penman-Monteith equation [53].

$$SEP_{(i)} = \frac{SR_{(i)} * e^{(-0.5*LAI_{(i)})} + 0.68 * 0.4 * VPD_{(i)}}{0.68 * Delt_{(i)}} \tag{3.11}$$

Where SR is the solar radiation, $Delt$ is slope of the saturation vapour pressure curve, LAI is daily leaf area index and VPD is daily vapour-pressure deficit, which is calculated from the minimum and maximum temperature.

$$VP_{(i)} = e^{20.386 - \frac{5132}{T_{max_{(i)}} \text{ or } min_{(i)} + 273}} \tag{3.12}$$

$$VPD_{(i)} = 0.75 * (VP_{(max)} - VP_{(min)}) \tag{3.13}$$

The slope of the saturation vapour pressure curve ($Delt$) is calculated in the following equation.

$$Delt_{(i)} = \frac{5336}{(T_{(i)} + 273)^2} * e^{(21.07 - \frac{5336}{(T_{(i)} + 273)})} \tag{3.14}$$

Where T is the temperature in Celsius.
In stage II, soil evaporation is triggered after the top soil layer has dried or when the transpirable soil of the entire soil is low or less than 0.5 [98].

$$SEP_{(i)} = SEP_{(i-1)} * (\sqrt{DSII_{(i)}} - \sqrt{DSII_{(i)} - 1}) \tag{3.15}$$

The daily transpiration rate (TR) is calculated from the biomass accumulation (PHS) as well as vapour pressure deficit. The transpiration rate can also be affected if the transpirable soil water on the top volume reduces.

$$TR1_{(i)} = \frac{PHS_{(i)} * VPD(i)}{0.09 * 1000} \tag{3.16}$$

$$TR_{(i)} = TR1_{(i)} * \left(\frac{2}{1 + e^{\frac{-14 * SW1_{(i)}}{19.5}}} - 1 \right) \tag{3.17}$$

Update the daily transpirable soil water in both volumes by considering the daily evaporation and crop transpiration.

$$SW1_{(i+1)} = \begin{cases} SW1_{(i)} - TR_{(i)} - SEP_{(i)} & SW1 > 0 \\ 0 & \text{Otherwise} \end{cases}$$

$$SW_{(i+1)} = \begin{cases} SW_{(i)} - TR_{(i)} - SEP_{(i)} & SW > 0 \\ 0 & \text{Otherwise} \end{cases}$$

The fraction of transpirable soil water ($FTSW$) for total soil volume is computed as follows:

$$FTSW_{(i+1)} = \frac{SW_{(i+1)}}{135} \tag{3.18}$$

3.1.3.2 Crop Nitrogen (N) Uptake

Over the last few years, we observed a significant increase in corn yield, which is essentially due to the efficient use of nitrogen in plants [17]. Nitrogen is an essential nutrient in determining final grain yield. Some of nitrogen's prominent roles in plant growth include stimulation of root development and activity and support of the uptake of other nutrients. These two roles significantly enhance water use efficiency. Nitrogen levels in corn range between 2.76 and 3.5%, while a level less than 2.25% N is considered insufficient.

Nitrogen in crops is determined using the supply-demand method but relatively constrains the variation in N concentrations in plant tissues. In other words, the parameters initially set for maximum and minimum tissue N content limit significant changes in physiological activity [99]. The uptake of N by plants is regulated either by plant requirements or nutrient soil supply.

[18][99][111] used an alternative approach for nitrogen supply calculation to the crop without considering a pre-established demand function. The characteristic of this approach is that the nitrogen uptake is determined independently, and the physiological activity is estimated by taking into account the resultant N levels in the crop tissue.

[99] shows a linear function between the accumulated heat unit (AHU) and the N uptake for maize. The total potential N uptake (PNU) in maize is estimated by considering the daily N uptake potential rate (NUP) and the heat-unit (HU). When the mineral N is insufficient in the soil, the daily N uptake (NU) is equal to the amount of N available in the soil for crop uptake [99].

$$NUP_{(i)} = \begin{cases} HU_{(i)} * \left\{ PNU * (5.24 * 10^{-8}) * AHU^{1.58} * e^{[-(AHU/958)^{2.58}]} \right\} & \text{SoilN} > 1 mgN/LH_2O \\ \text{SoilN} & \text{Otherwise} \end{cases}$$

N uptake depends on the fraction of transpirable soil water ($FTSW$); whenever $FTSW$ decreases to levels below a certain threshold, the value of NU considerably decreases.

$$NU_{(i)} = \frac{NUP_{(i)}}{1 + 9 * e^{(-15.3 * FTSW_{(i)})}} \tag{3.19}$$

In physiological development, applying inadequate N in maize results in significant depressed leaf area development, although leaf number is comparatively unaffected [100]. If the vegetation continues growing, the daily leaf nitrogen (LFN) estimates that the proportion of crop N uptake assigned to leaves ($PROPLFN$) is 0.6.

$$LFN_{(i+1)} = LFN_{(i)} + NU_{(i+1)} * PROPLFN \tag{3.20}$$

The Leaf Number (LN) is estimated as the ratio of cumulative LFN to leaf area index (LAI).

$$LN_{(i)} = \frac{LFN_{(i)}}{LAI_{(i)}} \tag{3.21}$$

If LN is less than the minimum LN (MLN), leaf area development will be inhibited; in this case MLN will be 0.55 gNm^{-2}.
According to [101], there is a linear relationship between LN and the radiation use efficiency (RUE), and it is referred to in the model as E.

$$RUE_{(i)} = \begin{cases} 0.12 + 1.09 * LN_{(i)} & LN < 1.35\ gNm^{-2} \\ 1.6 & \text{Otherwise} \end{cases}$$

The minimum LN of 0.55 gNm^2 resulted in a minimum RUE during leaf development.

In reproductive development, it is proven that maize produces grain with comparatively low N concentrations even though the N required to support seed growth is still significant.

In the maize model, the quantity of vegetative N available for translocation to seed (TVN) is determined at the beginning of grain growth. TVN refers to the total amount of N in the vegetative tissues at the beginning of seed growth over the minimum N contents for senesced leaves ($MSLN$) and mature stems ($MSTEMN$) [101].

$$TVN = f(MSLN, MSTEMN) \tag{3.22}$$

The daily total amount of N accumulated in the grain ($GRAINN$) is estimated as the sum of NU that is transferred from TVN.

$$GRAINN_{(i+1)} = GRAINN_{(i)} + NU_{(i)} + \frac{TVN * HU_{(i)}}{1150} \tag{3.23}$$

Where NU is the daily N uptake, TVN is the daily N translocation, and HU is the heat unit.

Since the grain biomass and N accumulation were estimated separately, the grain N concentration simulation also varied; however, two constraints were set on grain N concentration (GN) that might affect grain biomass and grain N accumulation. The same applies to leaf Nitrogen; if GN is less than the minimum GN, (MGN), grain development will be inhibited. In this case, MGN is 11 $gNKg^{-1}$.
The total grain N is estimated as follows

$$GN_{(i)} = \frac{GRAINN_{(i)}}{GRAIN_{(i)}} \tag{3.24}$$

Where ($GRAINN$) is the grain N accumulation and $GRAIN$ is the grain biomass.
If the grain N concentration reaches the maximum GN, N transfer to seeds will be restricted. In this case GN will be above 16 $gNKg^{-1}$. Consequently, the restricted N transfer will lead to a significant decrease in N loss in the leaves and stems.

$$GRAINN_{(i+1)} = GRAINN_{(i)} - \frac{TVN * HU_{(i)}}{3 * 1150} \tag{3.25}$$

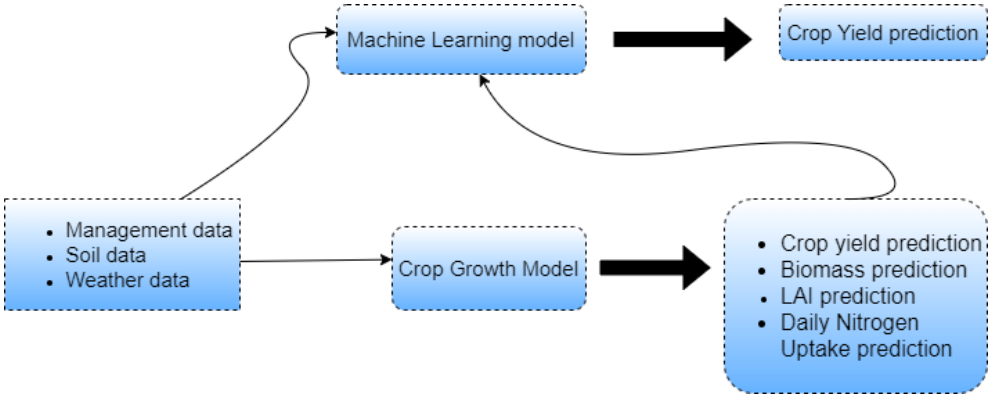


Figure 3.1: Theoretical Framework

3.2 Machine Learning Models

3.2.1 Theoretical Framework

Figure 3.1 presents the objective of this work. The goal is to combine machine learning and crop growth approaches. The historical data of the weather, soil and management will help evaluate the crop model and predict yield production, biomass, water and nitrogen stress.

The historical yield data are the target variables used to evaluate and enhance the yield predictions. The crop model outputs will help to train the ML models and forecast the final yield [2]. In this case, the input variables are the crop model outputs.

3.2.2 Performance Evaluation Measures

3.2.2.1 Dataset

The data was obtained from the USDA-NASS website [105] for the state of Iowa. The dataset includes information about corn yield, environment (soil, weather), and management (plant population). The environment and management data are considered as input variables and the corn yield as the target variable.

The historical data of corn yield was obtained from the USDA-NASS website for the year 1982 for a state in the US. Several factors affect crop yields, such as the environment, management, and genotype. Therefore, in this dataset, the weather and soil were included

as environmental features, while the plant population was included as a management feature.

- **Plant population**: One feature represents the yearly plant population and estimated plant state per square meter, acquired from the USDA-NASS website. Due to the unavailability of some data, we considered that the plant population for the simulation would be 7 plants m^{-2}.

- **Weather**: Weather is one of the most influential factors on yield outcome, compounded by the fact that we have no control over it. Extensive research and various technologies have designed mechanisms to help alleviate the impact of weather. Phenomena such as drought, high temperature, and extreme weather events are challenging to mitigate [108].

  Below are five weather variables accumulated daily from Daymet [107]

  1. Daily min air temperature in °C
  2. Daily max air temperature in °C
  3. Daily total precipitation in mL/day
  4. Daily solar radiation in MJ/m^2
  5. Daily vapour pressure in Pascal

- **Soil**: Soil components such as soil pH and organic matter, sand and clay content, soil bulk density, wilting point, field capacity, and saturation point, were taken into account. They can vary depending on the soil profile. A weighted average estimated different values for all layers. The data was acquired from the Web Soil Survey [106] for the location under study.

- **Nitrogen**: The total value of N fertilizer application varies between 11.6 g N m^{-2} and 40.1 g N m^{-2}. For the dataset, we used three levels of N applied (11.6, 27.6, and 40.1 g N m^{-2}) In [99, 111], the grain yield and N accumulation were well simulated for the 26.4 g N m^{-2} fertilizer treatment, and it reasonably represented the quantity of N in the soil available for crop uptake. In other words 26.4 g N m^{-2} of fertilizer was required to produce the maize grain yield.

- **Corn Yield**: Annual corn yield data in gram m^{-2}, obtained from USDA-NASS.

3.2.2.2 Data Pre-processing

Data pre-processing tasks are crucial to perform in ML because they help prepare the data before training them in different machine learning models.

Data Subset - The training data was divided into train and validation subsets; the latter was utilized to construct the models. The input variables were scaled to limit the large size of certain features confusing the ML models.

Feature Selection is conducted to circumvent overfitting in the training data due to its sparsity and many input variables.

We used principal component analysis (PCA), a well-known method in dimensionality-reduction, that is instrumental in processing data where multi-colinearity exists between the features. PCA lessens the number of dataset features by transforming a large set of features into a smaller one while maintaining as much information as possible [109].

The PCA can be defined following these four steps:

1. Determine the relationship among features through a Covariance Matrix.
2. Get eigenvectors and eigenvalues through the linear transformation or eigendecomposition of the Covariance Matrix.
3. Then, transform the data using Eigenvectors into principal components.
4. Finally, quantify the importance of these relationships using Eigenvalues and keep the important principal components.

Implementing PCA on the dataset helps to remove correlated variables to improve the algorithm's performance. Additionally, PCA aids in denoising and reducing overfitting in order to enhance visualization.

3.2.2.3 Model Selection and Hyperparameter

The hyperparameters of ML models must be calibrated, and then we select the best models to enhance prediction accuracy [2].

- Cross-Validation: Also known as k-fold cross-validation. Cross-validation is a technique that splits the training data to training and validation subsets. It helps find the best parameter values in order to reduce overfitting. Usually, a random tenfold cross-validation technique is used to set the machine learning models' hyperparameters and it is assumed to have the same size.
- Bayesian search: this method presumes the underlying distribution and approximates unknown factors by using substitute models such as the Gaussian process. The difference between Bayesian optimization and other types of search is the fact that it incorporates prior beliefs about underlying functions and adds new observations on top of that.

3.2.3 Predictive Models

We chose regression ML models to forecast crop output. Well-performing models require variation in base learner predictions.
A few parameters included in the dataset are yield, temperature, rainfall and solar radiation. Machine learning prediction will support farmers in choosing the most suitable parameters for optimal crop yield.
We use different metrics such as root mean square error, relative root mean square error, mean absolute error, and coefficient of determination to compare the selected machine learning models.

The machine learning selected and trained in this study consists of the random forest regressor, and artificial neural networks.

3.2.3.1 Random Forest Regressor

Random forest is a tree-based ensemble algorithm that can be employed in both classification and regression applications. In this work, we applied its regression tool. The RF model is based on the bootstrap aggregating (Bagging) method. Bagging is designed to decrease the variance of predictions and increase the average prediction of multiple trees by using a random sample with replacement [55].
The random forest model has some particularity to it, where each tree uses a random value of features in each split. The model averages all the predictions made by all trees to improve the accuracy and control the over-fitting. In this case, the random value of features will reduce the prediction errors and correlation of the trees. Thus, it will render the random forest model more efficient [56].

Random forest requires training every input feature to predict the yields of corn. To train RF models, multiple CART are built with an arbitrary subset of predictors with no pruning, and the average of the forest of CART is applied. The random forest algorithm is run in two different phases. In the first phase, a random forest is created, and in the second phase, a prediction is generated from the random forest produced in the first phase [27].
Below is the pseudo-code to create a random forest:

Algorithm 3.1 First phase: Creating the random forest

1. From a total of M features, choose K features at random so that K $\ll$ M
2. Define a node D contributing to the best split point amongst the K features.
3. Depending on the best split, divide D into several child nodes.
4. Repeat steps 1–3 until the proposed L nodes are attained.
5. Build the forest by repeating all steps.

Algorithm 3.2 Second phase: Resulting prediction from random forest

1. Apply pre-set rules for each randomly framed decision tree to obtain test features in predicting the target.
2. Identify votes for each predicted target.
3. Accept the predicted target with the greatest vote as the final prediction from the random forest algorithm.

A random forest algorithm that predicts the crop yield is built from the CART-based decision trees. Data is collected based on output from the CART model, and the ultimate prediction is reached using a group of CART trees, resulting in a random forest being created.

We used the random forest algorithm to estimate the values produced for the test data by evaluating $RMSE$, $RRMSE$, MAE, and R^2.

3.2.3.2 Neural Networks

Neural networks, also called artificial neural networks, are complex models that imitate human brain operation and consist of input, hidden and output layers [110]. The input layer consists of the soil and weather variables to predict the crop yield. In this case, we used the forward MLP to generate results [89, 97]. The algorithm is trained using the

backpropagation of error to modify the weights and decrease the errors, thereby achieving non-linear regression.

The neural network is built in two phases [27]:

Phase 1. Propagation

1. Generate the output values by propagating forward in the network.
2. Compute the cost to determine the error term.
3. Use propagation of the output activation function to evaluate difference between actualized and predicted values of output and hidden layers.

Phase 2: Update the weights

1. Determine the weight gradient – apply activation function (target-actual output).
2. Subtract a percentage of weight gradient from the weight.

The crop yield was predicted by using the environmental and weather parameters which compose the neurons. The hidden layer executes computations based on the rectified linear unit (ReLu) activation function. The obtained results are the corn yield forecasts based on test data training. The estimated values acquired for the test data using the MLP were estimated using the MAE.

Chapter 4

Experimental Results

These experiments will first simulate the mechanistic crop growth model to predict corn yield based on historical data. We will vary the value of the N fertilizer and select the value that maximizes grain yield.

Next, we will evaluate the performance of the machine learning algorithms in predicting corn yield. Lastly, we will optimize the model using the ML algorithm with the least prediction error to maximize corn yield in the state of Iowa.

4.1 Nitrogen Application

We evaluated the models with different N input values to estimate the impact of nitrogen on yield prediction. The values of N fertilizer applications vary between 11.6 g N m^{-2} and 40.1 g N m^{-2}. We used three different levels of applied N values including 11.6, 27.6, and 40.1 g N m^{-2}.

- **Low level of N**: When we apply 11.6 g N/m^2 of nitrogen, we get 565.26 g N/m^2 of grain yield, the accumulation of biomass is low, and the leaf area index (LAI) is reduced.
- **Intermediate level of N**: When we apply 27.6 g N/m^2 of nitrogen, we get 782.72 g N/m^2 of grain yield, the accumulation of biomass and leaf area index (LAI) are moderated.

- **High level of N**: When we apply 40.1 g N/m^2 of nitrogen, we get 712.44 g N/m^2 of grain yield. We can notice that as the level of N is high, the grain yield and accumulation of biomass are reduced, and the leaf area index (LAI) becomes very high. This also impacts the environment.

The highest grain yield 782.72 g N/m^2 was produced when we applied the intermediate level of N, 27.6 g N/m^2 fertilizer. The length of the grain growth period and leaf number remained unchanged.

Higher N levels increase the LAI, LFN, GRAINN and GN but decrease the grain yield.

4.2 Performance Metrics

The performance metrics are used to assess the performance of the implemented machine learning algorithms. Three different statistical performance metrics, RMSE, RRMSE and MAE, were used to estimate the error and the variance is explained by the (R^2) [27][64]

4.2.1 Root Mean Square Error (RMSE)

RMSE is a method that measures the standard deviation of errors predicted by a model in a dataset.

$$RMSE = \sqrt{\frac{\sum_i (y_i - \hat{y}_i)^2}{n}} \tag{4.1}$$

Where $\hat{y}_i$ is the predicted value, y_i, the actual value and n the number of observations.

4.2.2 Relative Root Mean Square Error (RRMSE)

RRMSE, also called the normalized root mean squared error, is the RMSE normalized by the mean of the actual values. As a rule, the lower the RRMSE values, the better.

$$RRMSE = \frac{RMSE}{\bar{y}} \tag{4.2}$$

4.2.3 Mean Absolute Error (MAE)

MAE measures the sum of the absolute difference between the predicted and the actual variables of an observation. It captures the average magnitude of errors in a group of predictions without accounting for their directions.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{4.3}$$

4.2.4 Coefficient of Determination (R^2)

The R^2 measures the difference in the dependent variable, which can be predicted using independent variables.

$$R^2 = \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2} \tag{4.4}$$

4.3 Experimental Setup

To run our experiments and construct the random forest model, we deduct that the quantity of trees is initially set as 10. The neural network is trained with 20 nodes, 1 hidden layer, a learning rate of 0.01, a rectified linear activation, linear activation function, and a stochastic gradient-based optimizer for the MLP model.

4.4 Numerical Results and Discussion

In what follows, we perform experiments by comparing the simulation results of MLP and random forest algorithms to predict corn yields. The purpose of the execution is to forecast a value close to the real value. The prediction performance of ML models varies because each model considers various aspects of the expected characteristics.

To define efficiency of the model, the variance between the predicted and mechanistic model values (i.e., errors) are evaluated using different error measures, namely, RMSE, RRMSE, MAE and R^2.

Machine Learning Model	Error Measures			
	RMSE (g/m^2)	RRMSE (%)	MAE (g/m^2)	R^2
Multilayer Perceptron	0.2423	0.4252	0.1586	0.7453
Random Forest	0.2470	0.4336	0.1327	0.7351

Table 4.1: Performance Metric Evaluation of the Models for Iowa

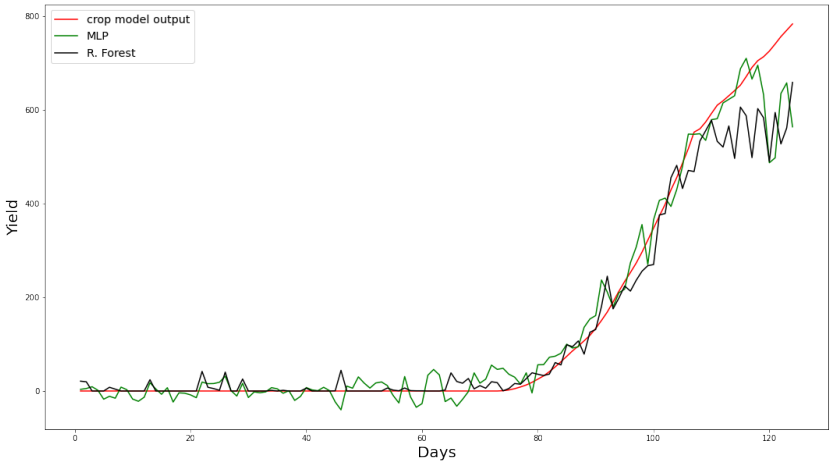


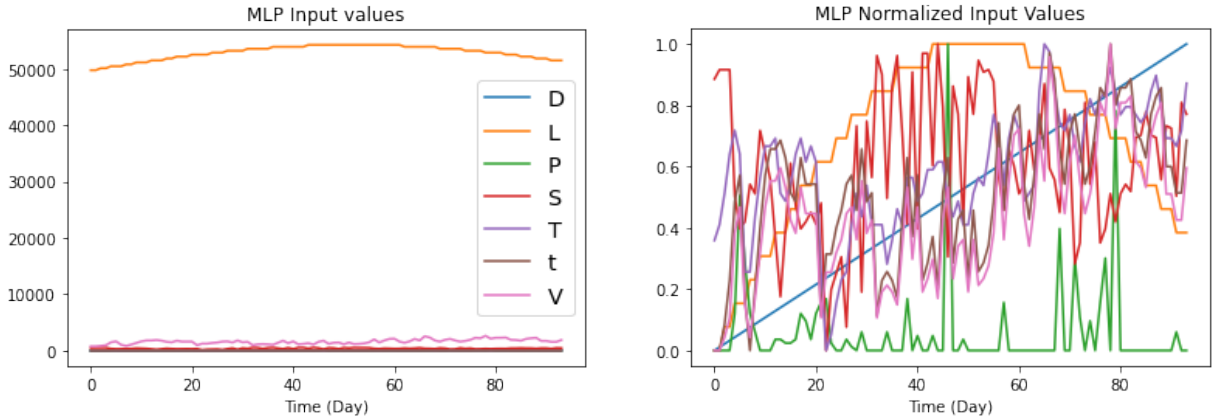
Figure 4.1: Annual Yield for Iowa

4.4.1 Results

In the following section, we summarize the results of the various experimental analyses of the ML models reviewed in Section 3.2.3

Table 4.1 shows acquired values for the test data of the state of Iowa, assessed using the following error measures: RMSE, RRMSE, MAE, and R^2.

Figure 4.1 shows the increase in corn yield over time. The MLP achieves the best performance and least error prediction compared to the random forest model.
The harvest is reached 86 days after planting. This state has the most ideal weather for growing corn and the most significant production of corn in the US.



(a) Actual Input

(b) Input normalized

Figure 4.2: Input Data

4.5 Model Optimization

In this section, we applied MLP methods of optimization to maximize crop yield using the N fertilizer amount of 27.6 g N m^{-2} in the state of Iowa.

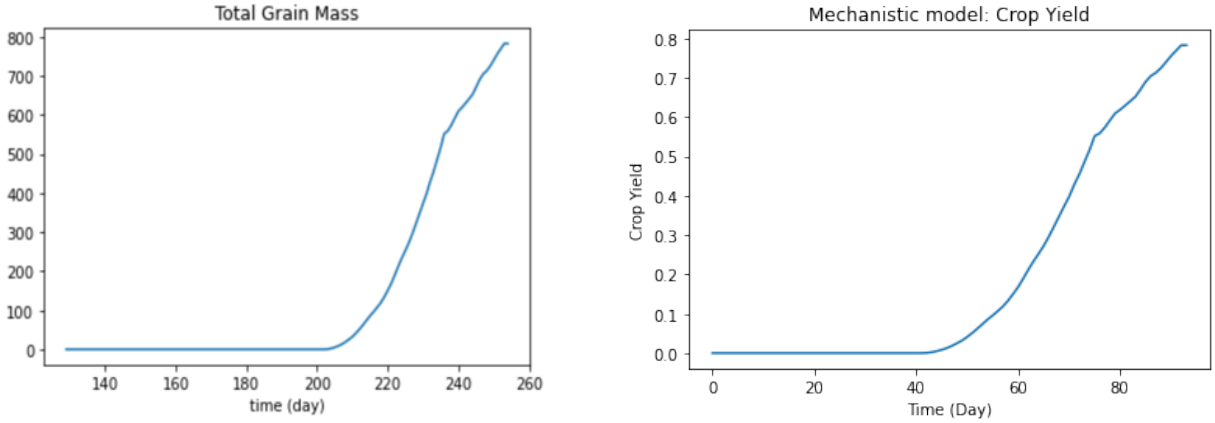
4.5.1 Normalization

After assessing the ideal ML model, MLP, it is necessary to optimize the factors that affect crop growth in order to maximize production.

MLP data preparation involves utilizing processes such as normalization to rescale input and output variables before training the MLP model. The dataset is rescaled in a range of $[0, 1]$ for the input value and the output value is divided by 1000 as shown in Figures 4.2 and 4.3.

We selected different parameters to optimize the model and consequently maximize corn yield prediction.

- **Input Layer**: this layer comprises the very beginning of the network, and is composed of seven different parameters.
- **Hidden Layer**: in this layer, a function applies weights to the inputs and leads them through an activation function as the output. In the simulation, we use one hidden layer.



(a) Actual Output (b) Output normalized

Figure 4.3: Output Data

- **Activation Function**: the hidden layer uses the rectified linear unit (ReLu) activation function, and the output layer uses the linear activation function.
- **Optimizer**: the Adam and SGD optimizers are used as the optimization technique to maximize output.
- **Output Layer**: this is the final layer of the network that outputs (produces) the prediction.

4.5.2 Optimizers

Optimizers are algorithms used to modify attributes such as the weights or learning rates of the neural network to reduce overall loss and improve accuracy. They also help to obtain results faster.

The choice of optimizers can significantly influence the performance of the model.

4.5.2.1 Adaptive moment estimation (Adam) Optimizer

Adam is an optimization algorithm that can update network weights iteratively based on training data. The algorithm is an efficient stochastic optimization with minimum memory requirements. It is highly suited for contending with large datasets and parameter

Scenarios	Adam Optimizer	SGD Optimizer
Actual dataset	0.865	0.896
Solar radiation adjusted	0.800	0.793
Precipitation adjusted	0.807	0.821
Minimum temperature adjusted	0.791	0.813
Maximum temperature adjusted	0.812	0.826
All selected parameters	0.820	0.888

Table 4.2: Results of Optimization with Adam and SGD Optimizers

problems, and issues with overly noisy and sparse gradients.
For different parameters, the algorithm calculates individual adaptive learning rates using assessments of the gradients' first and second moments [115].
The chosen settings for the tested MLP problems are $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$.

4.5.2.2 Stochastic Gradient Descent (SGD) Optimizer

SGD is an algorithm that is iterative in nature and used to optimize functions with suitable smoothness properties (e.g., differentiable). The algorithm removes the wait in the update and computes the parameter gradient using only one or a few training examples.
The algorithm first reduces the disparity in the parameter update and has potential to result in more stable convergence. Moreover, it lets the analysis benefit from greatly optimized matrix processes best applied in a well-vectorized calculation of the cost and gradient.
The chosen settings for the tested MLP problems are $\alpha = 0.01$, decay=0.0, momentum=0.7 and nesterov=False.

4.5.2.3 Model Simulation Scenarios

To perform the experiment, we executed different scenarios for the Adam and SGD optimizers to determine which one provides better optimization and maximizes grain yield. We first started with the actual dataset, and then adjusted parameters such as solar radiation, precipitation, and Min and Max temperature all together. Fig 4.4 and Table 4.2 show the optimization results with Adam and SGD optimizers.

The experiment results demonstrated that the highest grain 896 g/m^2 occurred when using the SGD optimizer with the dataset in the unchanged parameters scenario. Figure

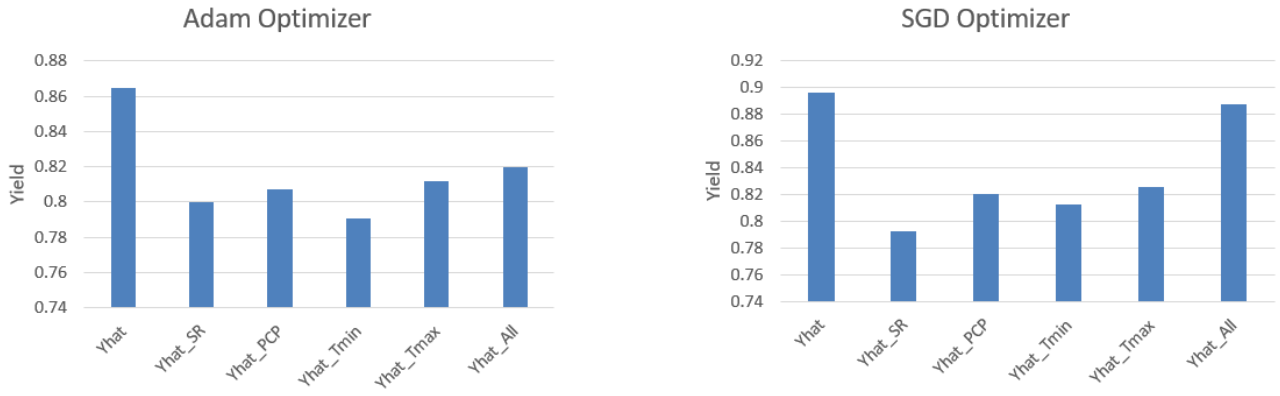


Figure 4.4: Results of Optimization with Adam and SGD Optimizers

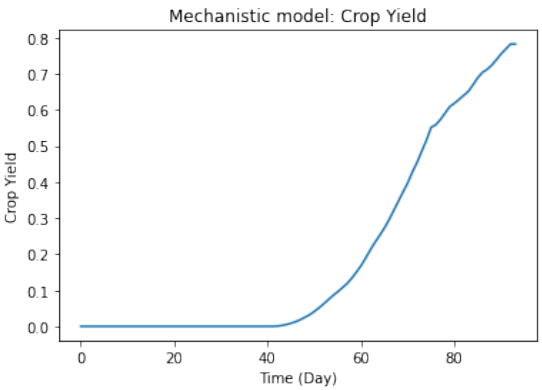
4.5 compares the grain yield obtained using the mechanistic model with the optimized grain yield using the MLP model.

In conclusion, the optimization of the model by using MLP-SGD optimizer provided a better prediction than the mechanistic model.

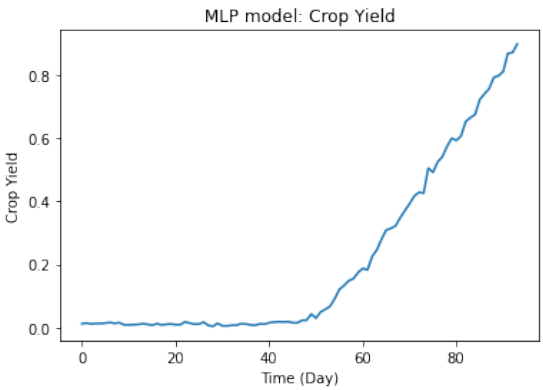
4.5.2.4 Learning Curve

A learning curve is a plot of model learning performance over experience or time.
Figure 4.6 shows a good fit of the learning curves since the plot of training loss decreases to the point of stability, and the validation loss decreases to the point of stability and has a small gap with the training loss. The learning performance has been done with the Epoch=100.
The MSE of training loss = 0.006 and validation loss = 0.007.

In conclusion, Figure 4.6 shows that the training and validation datasets are suitably representative.



(a) Actual Grain Yield



(b) Optimized Grain Yield

Figure 4.5: Comparison Between the Actual and Optimized Grain Yield

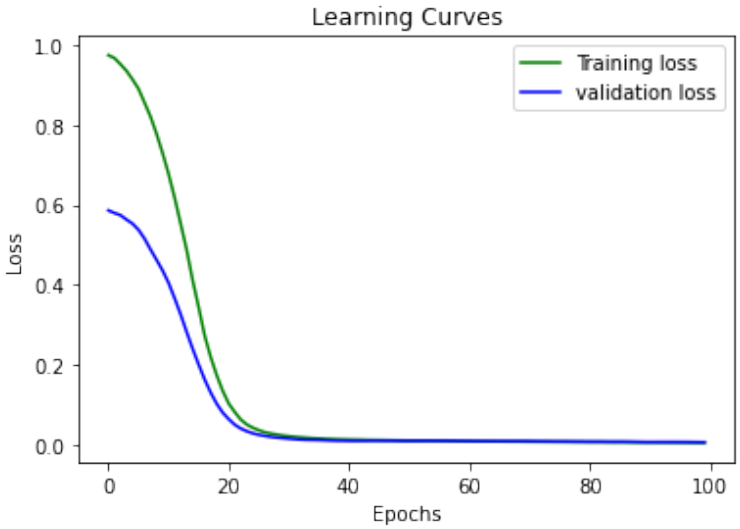


Figure 4.6: Learning Curves

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this thesis, we introduced a mechanistic crop growth model to predict corn growth and yield, and a machine learning model to optimize the predicted yield. The mechanistic crop growth model was built on parameters such as weather, soil management, and plant population data and reflects the physiological process of the growing season.
We evaluated three different applications of nitrogen levels to analyze their impact throughout the growing season. We observed that the levels of nitrogen applied had a significant impact on yield prediction.

Besides the historical data, we added the mechanistic crop growth model's output variables as inputs to the ML algorithms to investigate which combination (mechanistic crop growth model + ML predictive models) provided fewer prediction errors.
We evaluated ML algorithms such as multi-layer perceptron and random forest to perfect the combination. The performance of the algorithms was analyzed using evaluation metrics, including RMSE, RRMSE, MAE and R^2.
We then conducted a simulation to analyze the performance of two ML algorithms and selected the algorithm with the least prediction errors. Our results demonstrated that the multilayer perceptron algorithm achieved less performance errors in comparison to the random forest algorithm in terms of predicting corn yield.

In addition, we used the MLP model to optimize the dataset features. We performed various scenarios and applied two different optimization algorithms, Adam and SGD, to improve yield prediction, thus maximizing the yield.

Results showed that the optimization improve corn yield prediction compared to the mechanistic growth model. The highest grain yield occurred when using a stochastic gradient descent optimizer and the dataset with unchanged parameters scenario.

5.2 Future Work

In the future, we plan to work on four different schemes. First, we plan to investigate the feasibility of performing further data analytics on edge devices to determine its effectiveness in reducing excessive cloud resource exploitation for real-time applications in resource-constrained environments.
Second, we plan to run experiments on the Area X.0 datasets. This is important since it is local data and could help improve decision-making in smart farms located in Canada.
Third, we plan to involve various parameters such as crop disease, water salinity and pest control in the corn growth model and combine them with multiple ML algorithms to maximize the crop yield.
Fourth, we also plan to implement other use cases and optimization algorithms for various parameters involved in the crop growth model.

References

[1] B. Vandana, S. Kumar, "Hybrid K Mean Clustering Algorithm for Crop Production Analysis in Agriculture," International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, vol. Issue-2S, Dec. 2019

[2] M. Shahhosseini, G. Hu, and S. V. Archontoulis, "Forecasting Corn Yield With Machine Learning Ensembles," Frontiers in Plant Science, vol. 11, Jul. 2020, doi: 10.3389/fpls.2020.01120.

[3] N. Dlodlo and J. Kalezhi, "The internet of things in agriculture for sustainable rural development," 2015 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), 2015, pp. 13-18, doi: 10.1109/ETNCC.2015.7184801.

[4] S. Wolfert, L. Ge, C. Verdouw, and M.-J. Bogaardt, "Big data in smart farming—A review," Agricultural Systems, vol. 153, pp. 69–80, May 2017, doi: 10.1016/j.agsy.2017.01.023.

[5] O. Elijah, T. A. Rahman, I. Orikumhi, C. Y. Leow and M. N. Hindia, "An Overview of Internet of Things (IoT) and Data Analytics in Agriculture: Benefits and Challenges," IEEE Internet of Things Journal, vol. 5, no. 5, pp. 3758-3773, Oct. 2018, doi: 10.1109/JIOT.2018.2844296.

[6] A. Goap, D. Sharma, A. K. Shukla, and C. Rama Krishna, "An IoT based smart irrigation management system using Machine learning and open source technologies," Computers and Electronics in Agriculture, vol. 155, pp. 41–49, Dec. 2018, doi: 10.1016/j.compag.2018.09.040.

[7] G. Stamatescu, C. Drăgana, I. Stamatescu, L. Ichim and D. Popescu, "IoT-Enabled Distributed Data Processing for Precision Agriculture," 27th Mediter-

ranean Conference on Control and Automation (MED), 2019, pp. 286-291, doi: 10.1109/MED.2019.8798504.

[8] N. Kim, K.-J. Ha, N.-W. Park, J. Cho, S. Hong, and Y.-W. Lee, "A Comparison Between Major Artificial Intelligence Models for Crop Yield Prediction: Case Study of the Midwestern United States, 2006–2015," ISPRS International Journal of Geo-Information, vol. 8, no. 5, p. 240, May 2019, doi: 10.3390/ijgi8050240.

[9] D. Vasisht et al., "FarmBeats: An IoT Platform for Data-Driven Agriculture." This paper is included in the Proceedings of the 14th USENIX Symposium on Networked Systems Design and Implementation (NSDI '17). [Online]. Available: https://www.usenix.org/system/files/conference/nsdi17/nsdi17-vasisht.pdf

[10] C.A Jones, and J.R. Kiniry. CERES-maize: A Simulation Model of Maize Growth and Development. College Station, TX: Texas A & M Univ. Press, 1986.

[11] R. C. Muchow and P. S. Carberry, "Environmental control of phenology and leaf growth in a tropically adapted maize," Field Crops Research, vol. 20, no. 3, pp. 221–236, Apr. 1989, doi: 10.1016/0378-4290(89)90081-6.

[12] M. Stapper and G. F. Arkin, "Cornf: A Dynamic Growth and Development Model for Maize (Zea mays L.)," Program and Model Documentation No. 80-2, Texas A & M University, College Station, 1980.

[13] L. M. Dwyer and D. W. Stewart, "Leaf Area Development in Field-Grown Maize," Agronomy Journal, vol. 78, no. 2, pp. 334–343, 1986

[14] R. C. Muchow and P. S. Carberry, "Environmental control of phenology and leaf growth in a tropically adapted maize," Field Crops Research, vol. 20, no. 3, pp. 221–236, Apr. 1989, doi: 10.1016/0378-4290(89)90081-6.

[15] R. C. Muchow, T. R. Sinclair, and J. M. Bennett, "Temperature and Solar Radiation Effects on Potential Maize Yield across Locations," Agronomy Journal, vol. 82, no. 2, pp. 338–343, 1990

[16] J.L. Monteith, "Climate and the efficiency of crop production in Britain," Philosophical Transactions of the Royal Society of London. B, Biological Sciences, vol. 281, no. 980, pp. 277–294, Nov. 1977.

[17] J. T. Ritchie, "Model for predicting evaporation from a row crop with incomplete cover," Water Resources Research, vol. 8, no. 5, pp. 1204–1213, Oct. 1972, doi: 10.1029/wr008i005p01204.

[18] J. R. Williams, C. A. Jones, J. R. Kiniry, and D. A. Spanel, "The EPIC Crop Growth Model," Transactions of the ASAE, vol. 32, no. 2, pp. 0497–0511, 1989, doi: 10.13031/2013.31032.

[19] C. A. Jones, "A survey of the variability in tissue nitrogen and phosphorus concentrations in maize and grain sorghum," Field Crops Research, vol. 6, pp. 133–147, Jan. 1983, doi: 10.1016/0378-4290(83)90053-9.

[20] Y. Osakabe, K. Osakabe, K. Shinozaki, and L.-S. P. Tran, "Response of plants to water stress," Frontiers in Plant Science, vol. 5, Mar. 2014, doi: 10.3389/fpls.2014.00086.

[21] Y. Song, C. Birch, S. Qu, A. Dohert, and J. Hanan, "Analysis and Modelling of the Effects of Water Stress on Maize Growth and Yield in Dryland Conditions," Plant Production Science, vol. 13, no. 2, pp. 199–208, Jan. 2010, doi: 10.1626/pps.13.199.

[22] T. Chen, Precision: Principles, Practices and Solutions for the Internet of Things. San Jose, Ca: Crowdstory Publishing, 2016.

[23] S. K. Datta, C. Bonnet and N. Nikaein, "An IoT gateway centric architecture to provide novel M2M services," 2014 IEEE World Forum on Internet of Things (WF-IoT), 2014, pp. 514-519, doi: 10.1109/WF-IoT.2014.6803221.

[24] M. R. Anawar, S. Wang, M. Azam Zia, A. K. Jadoon, U. Akram, and S. Raza, "Fog Computing: An Overview of Big IoT Data Analytics," Wireless Communications and Mobile Computing, vol. 2018, pp. 1–22, 2018, doi: 10.1155/2018/7157192.

[25] K. A. Patil and N. R. Kale, "A model for smart agriculture using IoT," 2016 International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC), 2016, pp. 543-545, doi: 10.1109/ICGTSPICC.2016.7955360.

[26] S. Verma, R. Gala, S. Madhavan, S. Burkule, S. Chauhan and C. Prakash, "An Internet of Things (IoT) Architecture for Smart Agriculture," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018, pp. 1-4, doi: 10.1109/ICCUBEA.2018.8697707.

[27] D. Elavarasan, D. R. Vincent, V. Sharma, A. Y. Zomaya, and K. Srinivasan, "Forecasting yield by integrating agrarian factors and machine learning models: A survey," Computers and Electronics in Agriculture, vol. 155, pp. 257–282, Dec. 2018, doi: 10.1016/j.compag.2018.10.024.

[28] J. D. Kelleher, B. Mac Namee, and A. D'Arcy, "Fundamentals of machine learning for predictive analytics." Cambridge, MA; London: The MIT Press, 2015.

[29] R. S. Sutton and A. Barto, "Reinforcement learning : an introduction." Cambridge, MA ; London: The MIT Press, 2018.

[30] F. D. Whisler, B. Acock, D.N. Baker, R. E. Fye, "Crop Simulation Models in Agronomic Systems," Science Direct, Volume 40, Pages 141-208, 1986.

[31] V. R. K. Murthy, "Crop Growth modeling and its applications in agricultural meteorology.", Satellite Remote Sensing and GIS Applications in Agricultural Meteorology, pp. 235-261, 2004.

[32] X. Hu and S. Qian, "IOT application system with crop growth models in facility agriculture," 2011 6th International Conference on Computer Sciences and Convergence Information Technology (ICCIT), 2011, pp. 129-133.

[33] J. R. Williams and K. G. Renard, "Assessments of Soil Erosion and Crop Productivity with Process Models (EPIC)," Soil Erosion and Crop Productivity, R.F. Follett and B.A. Stewart, Ed. American Society of Agronomy, Inc., Crop Science Society of America, Inc., Soil Science Society of America, Inc., 1985, pp. 67–103 .

[34] X. C. Wang, J. Li, M. N. Tahir, and M. D. Hao, "Validation of the EPIC model using a long-term experimental data on the semi-arid Loess Plateau of China," Mathematical and Computer Modelling, vol. 54, no. 3, pp. 976–986, Aug. 2011, doi: 10.1016/j.mcm.2010.11.025.

[35] J.R. Williams, A.N. Sharpley, "EPIC-Erosion/Productivity Impact Calculator 1. Model Documentation." USD A Tech. Bulletin 1768, Ch. 2, 1990.

[36] T. Hodges, D. Botner, C. Sakamoto, and J. Hays Haug, "Using the CERES-Maize model to estimate production for the U.S. Cornbelt," Agricultural and Forest Meteorology, vol. 40, no. 4, pp. 293–303, Sep. 1987, doi: 10.1016/0168-1923(87)90043-8.

[37] M. Caprolu, R. Di Pietro, F. Lombardi, S. Raponi; "Edge Computing Perspectives: Architectures, Technologies, and Open Security Issues", 2019 IEEE International Conference on Edge Computing (EDGE), Milan, Italy, 8-13 July 2019.

[38] R. K. Kodali ; B. S. Sarjerao, "A low cost smart irrigation system using MQTT protocol" 2017 IEEE Region 10 Symposium (TENSYMP), Cochin, India, 14-16 July 2017.

[39] M. D. Donno, K. Tange, and N. Dragoni, “Foundations and Evolution of Modern Computing Paradigms: Cloud, IoT, Edge, and Fog”, in IEEE Access, vol. 7, pp. 150936-150948, 2019.

[40] M. Heydari ; A. Mylonas ; V. Katos ; E. Balaguer-Ballester; V. H. Fami Tafreshi; E. Benkhelifa, “Uncertainty-Aware Authentication Model for Fog Computing in IoT”, 2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC), Rome, Italy, 10-13 June 2019.

[41] G. Castellano; F. Risso; R. Loti, “Fog Computing over Challenged Networks: a Real Case Evaluation” 2018 IEEE 7th International Conference on Cloud Networking (CloudNet), 22-24 Oct. 2018.

[42] F. J. Ferra´ndez-Pastor, J. M. Garc´ıa-Chamizo, M. Nieto-Hidalgo and J. Mora-Mart´ınez, ”Precision Agriculture Design Method Using a Distributed Computing Architecture on Internet of Things Context”, Sensors, May 2018.

[43] M.A. Friedl, “Remote Sensing of Croplands”, Comprehensive remote sensing, CRC Press,Boca Raton, pp 78–95 DOI:10.1016/B978-0-12-409548-9.10379-3.

[44] J. Jayadevan, S.M. Jasmine, S. Kumar N “A Novel Architecture For Internet of Things in Precision Agriculture,” International Journal of Applied Engineering Research, vol. 15, no. 3, pp. 204-211, 2020.

[45] A. Boukhdhir, O. Lachiheb and M. S. Gouider, ”An improved mapReduce design of kmeans for clustering very large datasets,” 2015 IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA), 2015, pp. 1-6, doi: 10.1109/AICCSA.2015.7507226.

[46] A. B. Patel, M. Birla and U. Nair, ”Addressing big data problem using Hadoop and Map Reduce,” 2012 Nirma University International Conference on Engineering (NUiCONE), 2012, pp. 1-5, doi: 10.1109/NUICONE.2012.6493198.

[47] M. R. Bendre, R. C. Thool and V. R. Thool, ”Big data in precision agriculture: Weather forecasting for future farming,” 2015 1st International Conference on Next Generation Computing Technologies (NGCT), 2015, pp. 744-750, doi: 10.1109/NGCT.2015.7375220.

[48] S. Rajeswari, K. Suthendran and K. Rajakumar, ”A smart agricultural model by integrating IoT, mobile and cloud-based big data analytics,” 2017 International Conference on Intelligent Computing and Control (I2C2), 2017, pp. 1-5, doi: 10.1109/I2C2.2017.8321902.

[49] B. Omoniwa, R. Hussain, M. A. Javed, S. H. Bouk and S. A. Malik, "Fog/Edge Computing-Based IoT (FECIoT): Architecture, Applications, and Research Issues," in IEEE Internet of Things Journal, vol. 6, no. 3, pp. 4118-4149, June 2019, doi: 10.1109/JIOT.2018.2875544.

[50] P. Verma and S. K. Sood, "Fog Assisted-IoT Enabled Patient Health Monitoring in Smart Homes," in IEEE Internet of Things Journal, vol. 5, no. 3, pp. 1789-1796, June 2018, doi: 10.1109/JIOT.2018.2803201.

[51] H. Tianfield, "Towards Edge-Cloud Computing," 2018 IEEE International Conference on Big Data (Big Data), 2018, pp. 4883-4885, doi: 10.1109/BigData.2018.8622052.

[52] S. Liu, J. Y. Yang, C. F. Drury, H. L. Liu, and W. D. Reynolds, "Simulating maize (Zea mays L.) growth and yield, soil nitrogen concentration, and soil water content for a long-term cropping experiment in Ontario, Canada," Canadian Journal of Soil Science, vol. 94, no. 3, pp. 435–452, Aug. 2014, doi: 10.4141/cjss2013-096.

[53] R. Allen, L. Pereira, D. Raes, and M. Smith "Crop Evapotranspiration-Guidelines for Computing Crop Water Requirements-FAO Irrigation and Drainage", Paper 56; FAO: Rome, Italy, 1998; Volume 300, p. D05109.

[54] G.H. Hargreaves and Z.A. Samani, "Reference Crop Evapotranspiration from Temperature," Applied Engineering in Agriculture, vol. 1, no. 2, pp. 96–99, 1985, doi: 10.13031/2013.26773.

[55] L. Breiman, "Bagging predictors," Machine Learning, vol. 24, no. 2, pp. 123–140, Aug. 1996, doi: 10.1007/bf00058655.

[56] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/a:1010933404324.

[57] J. W. Jones et al., "The DSSAT cropping system model," European Journal of Agronomy, vol. 18, no. 3, pp. 235–265, Jan. 2003, doi: 10.1016/S1161-0301(02)00107-7.

[58] N. Balakrishnan and Dr. G. Muthukumarasamy, "Crop Production Ensemble Machine Learning Model for Prediction," International Journal of Computer Science and Software Engineering (IJCSSE), vol. 5, Issue 7, pp. 148-153, July 2016.

[59] M. Kavre, A. Gadekar and Y. Gadhade, "Internet of Things (IoT): A Survey," 2019 IEEE Pune Section International Conference (PuneCon), 2019, pp. 1-6, doi: 10.1109/PuneCon46936.2019.9105831.

[60] C. Bell, MySQL for the Internet of Things. Berkeley, CA: Apress, 2016. doi: 10.1007/978-1-4842-1293-6.

[61] H. Hu, Y. Wen, T.-S. Chua, and X. Li, "Toward scalable systems for big data analytics: A technology tutorial," IEEE access, vol. 2, pp. 652–687, 2014.

[62] A. De Mauro, M. Greco, M. Grimaldi, "A Formal Definition of Big Data Based on its Essential Features," Library Review, vol. 65, 122–135, 2016.

[63] M. D´ıaz, C. Mart´ın, and B. Rubio, "State-of-the-art, challenges, and open issues in the integration of internet of things and cloud computing," Journal of Network and Computer Applications, vol. 67, pp. 99–117, 2016.

[64] M. Shahhosseini, G. Hu, I. Huber, and S. V. Archontoulis, "Coupling machine learning and crop modeling improves crop yield prediction in the US Corn Belt," Scientific Reports, vol. 11, no. 1, p. 1606, Jan. 2021, doi: 10.1038/s41598-020-80820-1.

[65] M. Marwan, M. Fuad, P.-F. Marteau,"Towards a faster symbolic aggregate approximation method", Fifth International Conference on Software and Data Technologies, Athens, Greece, 2010.

[66] C. T. Zan and H. Yamana, "An improved symbolic aggregate approximation distance measure based on its statistical features," Proceedings of the 18th International Conference on Information Integration and Web-based Applications and Services, Nov. 2016, doi: 10.1145/3011141.3011146.

[67] Z. Wu, K. Qiu, and J. Zhang, "A Smart Microcontroller Architecture for the Internet of Things," Sensors, vol. 20, no. 7, p. 1821, Mar. 2020, doi: 10.3390/s20071821.

[68] N. R. Patel, P. D. Kale, G. N. Raut, P. G. Choudhari, N. R. Patel and A. Bherani, "Smart design of microcontroller based monitoring system for agriculture," 2014 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2014], 2014, pp. 1710-1713, doi: 10.1109/ICCPCT.2014.7054949.

[69] http://www.iot.qa/2018/01/iot-applications-in-agriculture_23.html

[70] E. Ahmed et al., "The role of big data analytics in Internet of Things," Computer Networks, vol. 129, pp. 459–471, Dec. 2017, doi: 10.1016/j.comnet.2017.06.013.

[71] M. S. Farooq, S. Riaz, A. Abid, K. Abid and M. A. Naeem, "A Survey on the Role of IoT in Agriculture for the Implementation of Smart Farming," in IEEE Access, vol. 7, pp. 156237-156271, 2019, doi: 10.1109/ACCESS.2019.2949703.

[72] N. G. Rezk, E. E.-D. Hemdan, A.-F. Attia, A. El-Sayed, and M. A. El-Rashidy, "An efficient IoT based smart farming system using machine learning algorithms," Multimedia Tools and Applications, vol. 80, no. 1, pp. 773–797, Sep. 2020, doi: 10.1007/s11042-020-09740-6.

[73] A. Nayyar and V. Puri, "Smart farming: IoT based smart sensors agriculture stick for live temperature and moisture monitoring using Arduino, cloud computing & solar technology," Research Gate, Nov. 09, 2016.

[74] Z. Yang, Y. Yue, Y. Yang, Y. Peng, X. Wang and W. Liu, "Study and application on the architecture and key technologies for IOT," 2011 International Conference on Multimedia Technology, 2011, pp. 747-751, doi: 10.1109/ICMT.2011.6002149.

[75] Wu, M.; Lu, T.J.; Ling, F.Y.; Sun, J.; Du, H.Y. "Research on the architecture of Internet of Things". In Proceedings of the 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), Chengdu, China, 20–22 August 2010; Volume 5, pp. V5-484–V5-487.

[76] F. Ferrández-Pastor, J. García-Chamizo, M. Nieto-Hidalgo, J. Mora-Pascual, and J. Mora-Martínez, "Developing Ubiquitous Sensor Network Platform Using Internet of Things: Application in Precision Agriculture," Sensors, vol. 16, no. 7, p. 1141, Jul. 2016, doi: 10.3390/s16071141.

[77] N. Sastry and D. Wagner, "Security considerations for ieee 802.15. 4 networks," in Proceedings of the 3rd ACM workshop on Wireless security. ACM, 2004, pp. 32–42.

[78] M. Tao, X. Hong, C. Qu, J. Zhang and W. Wei, "Fast access for ZigBee-enabled IoT devices using raspberry Pi," 2018 Chinese Control And Decision Conference (CCDC), 2018, pp. 4281-4285, doi: 10.1109/CCDC.2018.8407868.

[79] "Digitisation in agriculture - from precision farming to farming 4.0", Bioeconomy", Apr. 09, 2018. https://www.biooekonomie-bw.de/en/articles/dossiers/digitisation-in-agriculture-from-precision-farming-to-farming-40

[80] A. Walter, R. Finger, R. Huber, and N. Buchmann, "Opinion: Smart farming is key to developing sustainable agriculture," Proceedings of the National Academy of Sciences, vol. 114, no. 24, pp. 6148–6150, Jun. 2017, doi: 10.1073/pnas.1707462114.

[81] J. Budakoti, "An IoT Gateway Middleware for Interoperability in SDN Managed Internet of Things," Ph.D. dissertation, Carleton University, 2018.

[82] F. Bonomi, R. Milito, P. Natarajan, and J. Zhu, Big Data and Internet of Things: A Roadmap for Smart Environments. N. Bessis and C. Dobre, Eds. Cham: Springer International Publishing, 2014. doi: 10.1007/978-3-319-05029-4.

[83] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in Proceedings of the first edition of the MCC workshop on Mobile cloud computing. ACM, 2012, pp. 13–16.

[84] M. S. Obaidat and P. Nicopolitidis, Smart cities and homes: Key enabling technologies. Morgan Kaufmann, 2016.

[85] L. Martínez-Ferrer, M. Piles and G. Camps-Valls, "Crop Yield Estimation and Interpretability With Gaussian Processes," in IEEE Geoscience and Remote Sensing Letters, vol. 18, no. 12, pp. 2043-2047, Dec. 2021, doi: 10.1109/LGRS.2020.3016140.

[86] K. Kuwata and R. Shibasaki, "Estimating crop yields with deep learning and remotely sensed data," 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2015, pp. 858-861, doi: 10.1109/IGARSS.2015.7325900.

[87] N. Kim and Y.-W. Lee, "Machine Learning Approaches to Corn Yield Estimation Using Satellite Images and Climate Data: A Case of Iowa State," Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography, vol. 34, no. 4, pp. 383–390, Aug. 2016, doi: 10.7848/ksgpc.2016.34.4.383.

[88] S. Khanal, J. Fulton, A. Klopfenstein, N. Douridas, S. Shearer, "Integration of high resolution remotely sensed data and machine learning techniques for spatial prediction of soil properties and corn yield", Computers and Electronics in Agriculture, Volume 153, Pages 213-225, October 2018

[89] S. S. Panda, D. P. Ames, and S. Panigrahi, "Application of Vegetation Indices for Agricultural Crop Yield Prediction Using Neural Network Techniques," Remote Sensing, vol. 2, no. 3, pp. 673–696, Mar. 2010, doi: 10.3390/rs2030673.

[90] S. Andriyas and M. McKee, "Recursive partitioning techniques for modeling irrigation behavior," Environmental Modelling & Software, vol. 47, pp. 207–217, Sep. 2013, doi: 10.1016/j.envsoft.2013.05.011.

[91] B. A. Keating et al., "An overview of APSIM, a model designed for farming systems simulation," European Journal of Agronomy, vol. 18, no. 3, pp. 267–288, Jan. 2003, doi: 10.1016/S1161-0301(02)00108-9.

[92] D. P. Holzworth et al., "APSIM – Evolution towards a new generation of agricultural systems simulation," Environmental Modelling & Software, vol. 62, pp. 327–350, Dec. 2014, doi: 10.1016/j.envsoft.2014.07.009.

[93] L.M. Giraldo, L.J. Lizcano, A.J. Gijsman, B. Rivera, L.H. Franco, "Adaptation of the DSSAT model for simulation of Brachiaria decumbens production," Pasturas Tropicales 20, 2/12, 1998.

[94] A. de Wit et al., "25 years of the WOFOST cropping systems model," Agricultural Systems, vol. 168, pp. 154–167, Jan. 2019, doi: 10.1016/j.agsy.2018.06.018.

[95] N. J. Hadiya, N. Kumar, and B. M. Mote, "Use of WOFOST model in agriculture- A review," Agricultural Reviews, vol. 39, no. 3, pp. 234–240, Jul. 2018.

[96] Z. Cheng, J. Meng, and Y. Wang, "Improving Spring Maize Yield Estimation at Field Scale by Assimilating Time-Series HJ-1 CCD Data into the WOFOST Model Using a New Method with Fast Algorithms," Remote Sensing, vol. 8, no. 4, p. 303, Apr. 2016, doi: 10.3390/rs8040303.

[97] M. Kaul, R. L. Hill, and C. Walthall, "Artificial neural networks for corn and soybean yield prediction," Agricultural Systems, vol. 85, no. 1, pp. 1–18, Jul. 2005, doi: 10.1016/j.agsy.2004.07.009.

[98] J. Amir and T. R. Sinclair, "A model of water limitation on spring wheat growth and yield," Field Crops Research, vol. 28, no. 1–2, pp. 59–69, Dec. 1991, doi: 10.1016/0378-4290(91)90074-6.

[99] T. R. Sinclair and R. C. Muchow, "Effect of Nitrogen Supply on Maize Yield: I. Modeling Physiological Responses," Agronomy Journal, vol. 87, no. 4, pp. 632–641, Jul. 1995, doi: 10.2134/agronj1995.00021962008700040005x.

[100] J. H. Lemcoff and R. S. Loomis, "Nitrogen Influences on Yield Determination in Maize 1," Crop Science, vol. 26, no. 5, pp. 1017–1022, Sep. 1986, doi: 10.2135/cropsci1986.0011183x002600050036x.

[101] R. C. Muchow and R. Davis, "Effect of nitrogen supply on the comparative productivity of maize and sorghum in a semi-arid tropical environment II. Radiation interception and biomass accumulation," Field Crops Research, vol. 18, no. 1, pp. 17–30, Feb. 1988, doi: 10.1016/0378-4290(88)90056-1.

[102] H. Chen, W. Wu, and H.-B. Liu, "Assessing the relative importance of climate variables to rice yield variation using support vector machines," Theoretical and Applied Climatology, vol. 126, no. 1–2, pp. 105–111, Jul. 2015, doi: 10.1007/s00704-015-1559-y.

[103] T. Capehart and S. Proper, "Corn is America's Largest Crop in 2019," Usda.gov, 2019. [Online]. Available: https://www.usda.gov/media/blog/2019/07/29/corn-americas-largest-crop-2019

[104] B. Dumont, B. Basso, V. Leemans, B. Bodson, J.-P. . Destain, and M.-F. . Destain, "A comparison of within-season yield prediction algorithms based on crop model behaviour analysis," Agricultural and Forest Meteorology, vol. 204, pp. 10–21, May 2015, doi: 10.1016/j.agrformet.2015.01.014.

[105] USDA NASS. Surveys. National Agricultural Statistics Service, U.S. Department of Agriculture, 1982.

[106] Soil Survey Staff, Natural Resources Conservation Service, United States Department of Agriculture, Web Soil Survey, 1982.

[107] M.M. Thornton , R. Shrestha, Y. Wei, P.E. Thornton, S. Kao, and B.E Wilson 2020. Daymet: Daily Surface Weather Data on a 1-Km Gride for North America, Version 4, ORNL DAAC, Oak Ridge, Tennessee, USA. daac.ornl.gov. https://daac.ornl.gov/DAYMET/guides/Daymet_Daily_V4.html

[108] R. P. Motha, "The Impact of Extreme Weather Events on Agriculture in the United States", Challenges and Opportunities in Agrometeorology, pp. 397–407, 2011, doi: 10.1007/978-3-642-19360-6-30.

[109] K. Pothuganti, "Overview on principal component analysis algorithm in machine learning," International research journal of science and technology, Oct. 2020.

[110] J. A. Feldman, M. A. Fanty and N. H. Goodard, "Computing with structured neural networks," in Computer, vol. 21, no. 3, pp. 91-103, March 1988, doi: 10.1109/2.34.

[111] J.M. Bennett, L.S.M. Mutti, P.S.C. Rao and J.W. Jones, "Interactive effects of nitrogen and water stresses on biomass accumulation, nitrogen uptake, and seed yield of maize," Agronomy Physiology Laboratory, University of Florida, Gainesville, FL 32611 U.S.A, 1988

[112] "Multilayer Perceptron," DeepAI, May 17, 2019. https://deepai.org/machine-learning-glossary-and-terms/multilayer-perceptron

[113] R. E. Baker, J.-M. Peña, J. Jayamohan, and A. Jérusalem, "Mechanistic models versus machine learning, a fight worth fighting for the biological community?," Biology Letters, vol. 14, no. 5, p. 20170660, May 2018, doi: 10.1098/rsbl.2017.0660.

[114] K. Jorner, T. Brinck, P.-O. Norrby, and D. Buttar, "Machine learning meets mechanistic modelling for accurate prediction of experimental activation energies," Chemical Science, vol. 12, no. 3, pp. 1163–1175, Jan. 2021, doi: 10.1039/D0SC04896H.

[115] D.P. Kingma, J. Ba, "ADAM: A Method for Stochastic Optimization," ICLR, San Diego, USA, 2015