Automated

Precise Irrigation System.

Abstract:

Efficient irrigation practices are crucial for maximizing agricultural productivity and preserving water resources. Traditional irrigation methods are effective in areas with abundant water, but they fall short in regions facing water scarcity. To address this challenge, we propose a novel approach called IoT-driven root zone injection method of irrigation, which leverages machine learning, IoT devices, wireless neural networking of sensors, and root zone injection equipment for automation.

The process begins by collecting real-time data from the agricultural field using a wireless neural network of sensors, specifically tailored to the crop type being cultivated. This data is then used to create a comprehensive dataset. To ensure accurate predictions, the dataset undergoes pre-processing and cleaning before being fed into machine learning algorithms. In our case, we employ linear regression to estimate the required water content for irrigation on a given day.

The estimation of water content is dynamic and accounts for the varying needs of plants throughout their growth stages. For instance, water requirements are minimal during the initial phase, peak during the middle stage, and may vary depending on the plant species during the later stages. By incorporating these considerations, our approach achieves precise water content estimation.

To deliver the estimated water content to the plants, we rely on IoT devices that have been programmed with irrigation procedures. The machine learning predictions serve as a guide to the IoT system, informing it about the appropriate amount of water to be delivered. The system then initiates the irrigation process, with a key distinction being the use of root zone injection. This method ensures that water is directly supplied to the underground root zones of the plants.

By adopting root zone injection, our approach minimizes wastage due to evaporation, as water is delivered precisely where it is needed. This targeted approach significantly improves the efficiency of the irrigation process, leading to optimal resource utilization. Ultimately, our proposed method enables effective irrigation even in water-scarce areas, enhancing crop yield while conserving precious water resources.

1. Introduction:

Water scarcity has intensified in recent years due to the escalating global warming caused by greenhouse effects, leading to catastrophic changes. These alterations have disrupted rainfall patterns in rural, semi-arid, and urban areas, resulting in a prevalent water scarcity crisis and widespread droughts in agricultural regions. Consequently, there is an urgent need to conserve water resources and safeguard agricultural practices and land. However, water conservation alone is inadequate; optimizing irrigation practices is essential for achieving higher crop yields. Alarmingly, agriculture consumes around 85% of the Earth's available freshwater, underscoring the importance of efficient agricultural practices. Irrigation, among various factors such as fertilizers, climate, soil type, and plant selection, plays a crucial role as it directly impacts water resource utilization.

In water-scarce areas, traditional irrigation methods are inefficient in utilizing available water resources and conserving water for future use. Multiple factors influence irrigation practices, including humidity, plant growth phase, soil nutrition, plant species, root depth, soil type, temperature, and climate. While wireless sensor networks can monitor temperature and humidity, accurately estimating root depth remains challenging as it is hidden underground. Hence, alternative methods are needed to estimate root depth precisely, as it directly impacts irrigation practices focused on the root zone.

Machine learning, a recent technological advancement, offers powerful tools for real-time classification, clustering, and prediction tasks by analysing historical datasets. Regression-based prediction models in machine learning can estimate the continuous value of root depth for specific plant types over time. These models leverage data collected through wireless sensor networks, which have proven highly useful. By considering the time since planting, the prediction model estimates the root depth value, enabling determination of the required water content for irrigation on each cultivation day. Since water requirements fluctuate throughout the agricultural cycle until harvest, irrigation must be adjusted accordingly. Machine learning facilitates the prediction of changing water content values, providing valuable insights for irrigation planning from planting to harvest.

The Root Zone Injection (RII) method, also known as subsurface irrigation, is an emerging approach that directly delivers irrigation water to plant root zones beneath the soil surface. This method employs subsurface infiltration-promoting apparatuses (SIPA) for efficient water delivery. Injection nozzles are drilled into the soil near the root zones, enabling water to be directed to the underground root system. The depth at which the nozzle is placed depends on the phase of root growth and the estimated root depth derived from the ML prediction model. The irrigation system's precise control is achieved through the integration of Arduino Uno boards, XBee Modules, and Relay Modules, working in conjunction with sensors and the ML model. This setup governs the water supply to the plants through the RII method, optimizing water usage, enhancing efficiency, and ensuring accurate irrigation practices throughout the cultivation process.

1. Conceptual Review:

This section examines automated precision irrigation systems that employ various machine learning methods, along with intelligent technologies such as IoT and wireless neural networks, to promote conservative irrigation practices and enhance agricultural yields.

Yan-Ping Wang introduced a new irrigation method called root zone injection irrigation (RII), which has proven to be more efficient than traditional surface drip irrigation (SDI) in water-scarce areas. The RII method utilizes subsurface infiltration-promoting apparatuses (SIPA) that are drilled into the soil within the 0-0.6-meter layer, targeting the concentrated root zone of apple orchards. Testing this method in an apple orchard for three years revealed that the RII method consistently maintained water content above 60% of the field capacity, surpassing traditional SDI. Consequently, the proposed system achieved superior irrigation efficiency and water-use efficiency compared to other irrigation modes.

Chiyurl Yoon proposed an agricultural IoT system that combines low-power Bluetooth and low wide area networks (LPWAN) with existing wired communication networks based on Arduino technology. This system utilizes the MQ Telemetry Transport (MQTT) communication method, an IoT-dedicated protocol, to implement monitoring and control functions. By integrating existing and new technologies, this proposed system aims to reduce maintenance costs of existing devices while ensuring compatibility with new devices.

Patil K. A., Kale N. R. put forth an automated data collection and forecasting agriculture system consisting of three modules: Farm side, Server side, and Client side. This system utilizes wireless communication, Remote Monitoring System (RMS), and the internet to collect real-time data on the agriculture production environment. The collected data provides facilities such as alerts via short message service and advice on weather patterns and crop yield. With six modules encompassing sensing local agricultural parameters, data collection and transmission, decision-making support and warnings through data analysis, actuation and control, and crop monitoring via a camera module, this system establishes the framework for initial machine learning-based data collection and analysis in the agricultural field.

Rajinder Kumar Math proposed a framework for precision agriculture utilizing IoT. The system employs low-cost environmental sensors, Arduino Uno boards, wireless transceivers (XBee ZB S2), and actuating circuits to enable automated irrigation. ZigBee technology, built on the IEEE 802.15.4 standard, enables real-time data collection of parameters such as humidity, moisture content, and temperature for proper plant growth and automated irrigation. With the advancements in IoT, this system optimizes resource usage by providing precision irrigation in the precise quantity required by the crops, thereby reducing water wastage.

Harmantoa conducted experiments using four different levels of drip irrigation equivalent to 25%, 50%, 75%, and 100% of evapotranspiration based on the Penman-Monteith method. The study examined the effects on crop growth, yield, and water productivity using continuous and intermittent irrigation modes. Tomatoes were grown in a greenhouse, and performance was evaluated using distribution uniformity, emitter flow rate, and pressure head for various discharge rates. These experimental findings are utilized in the implementation chapter for estimating the required water content for crop cultivation.

Bright Keswani proposed an adaptive automated irrigation system that responds to weather conditions. This precision agriculture model employs an independent wireless sensor network comprising soil moisture probes, soil and environmental temperature sensors, humidity sensors, and daylight intensity sensors for real-time data collection through multi-point measurement. The acquired farm data is used to generate necessary actions throughout the entire farming period. It employs a structural similarity index (SSIM)-based water valve management system to ensure precision irrigation.

1. System Modules:

The automated precise irrigation system is composed of three modules.

First Module: The first module focuses on wireless sensor network data collection. In this module, farming data is gathered from various sensors, including humidity, moisture, temperature, climatic conditions, and soil texture. The collected data is initially in a raw format, encompassing a wide range of formats and scopes. Subsequently, the data is processed and refined to construct a dataset suitable for the prediction model. The resulting dataset is prepared in a format that can be seamlessly integrated into the second module. The intricacies of the data collection process, data cleaning, dataset formatting, data feeding, and model construction will be elaborated upon in the dataflow section.

Second Module: In order to estimate the root depth of farmed crops, the second module's machine learning component uses the dataset created in the first module along with information on the crop's botanical makeup and user input on the day it was planted. Once the dataset has been fed into machine learning methods like linear regression and random forest, a prediction model will be created that will allow for the estimation of the root depth of the chosen crop. Predicting the root depth is a crucial component of the proposed system since water content estimation will be done using both the root depth prediction value and field data collected using WSN. As a result, to anticipate the water content needed for precision irrigation, a secondary model is developed using the output of the primary model and data gathered from the field.

Third Module: The actual irrigation will be carried out in the third module using IoT devices like Arduino and valve-controlling commands processed by fuzzy logic. The Internet of Things (IoT) devices read the final estimated value of the water content need for that specific day and time period and start the irrigation process by managing water valves based on structural similarity (SSIM). Finally, the water from neutron probes reaches the RII system injection nozzle, which is drilled into the soil surface at the point where the root zone is strongly concentrated and which supplies water to the root zones of agricultural crops directly. By using dynamic root depth prediction and irrigation throughout the agricultural timeline, it is possible to achieve precision irrigation with zero evaporation, little water waste, and maximum resource utilisation. All three of these modules are shown together in Fig. 1 as the full proposed system design.

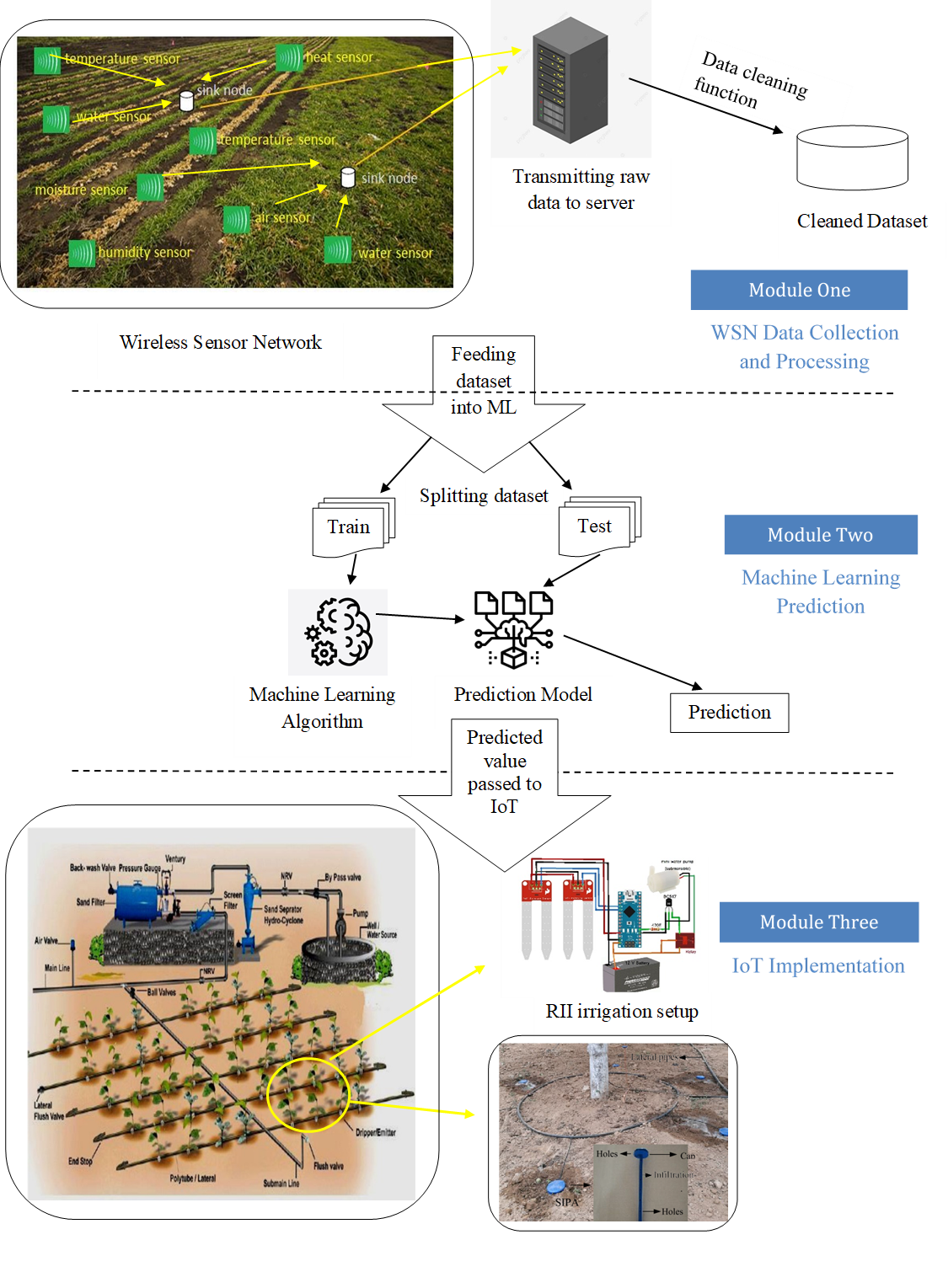


Figure 1: System Architecture

1. Implementation/Incorporation:

The three primary modules listed in architecture Fig. 1 are used in conjunction with a number of sub-modules in each component of the proposed system's implementation, which is carried out in stages. As a result, the general clear explanation of how the proposed system will operate is provided in the following manner in this section:

4.1 Networks of Wireless Sensors

The WSN is a collection of widely dispersed sensors that monitors, records, and stores atmospheric physical changes on a central server. Any metric value may be the environmental changes. There are two distinct nodes that define how the WSN functions. The sink nodes are in charge of accepting data from the sensing nodes and transmitting the raw data to the processing area. Sensing nodes are task-specific sensors that constantly track physical changes in the environment. The two ways for performing this data collecting and transmission in a wireless sensor network are referred to as single hop and multi hop data transmission. Figure 2 describes it.

Wireless Sensor Node



Fig 2: Single hop and Multi hop data transmission



**Sink**

**Sink**

In Figure 2, a cluster of sensors is represented by dotted circles, and the cluster head (CH) is represented by the black nodes in the cluster. When data is transmitted in a single hop, it is sent directly from the cluster head to the sink without any intermediate transmissions. However, when data is transmitted in a multi hop, the cluster head (CH) can send the data to the sink through a number of intermediate transmissions to other cluster heads before it reaches the sink node. When there is a long transmission distance, this multi hop transmission method is very helpful.

This method uses both single-hop and multi-hop transmission depending on the amount of ground that needs to be covered. Single-hop is an effective protocol to transport data to a farm land with a relatively limited area to cover since the transmission distance is short and the network model is much easier to implement.

However, because the network is characterised by its sensors' restricted wireless channel capacity, multi-hop transmission is a great option when there is a very large area to cover. As a result, single-hop transmission will require a lot of energy, whereas multi-hop transmission, which uses numerous intermediate transfers, dramatically reduces energy consumption via intricately connected data transmission networks.

When using direct communication, each sensor node must communicate its data directly to the cluster head or sink, and if the distance is great, this protocol will quickly deplete the CH battery and lifespan. Thus, direct communication only becomes an effective option in closer circumstances or with more battery power. However, because of its long-distance features, it is possible for data packets (collected data) to be lost. Therefore, cooperative communication, where the loss of data packets is handled, is a superior option for long-distance transfers.

The packet reception rate (PRR) over the transmission distance of “d” is determined by,

Based on the packet reception rate for a particular node as mentioned in eq (1), the neighbourhood set G(u) of that particular node can be given as,

According to equation (2), the subsequent neighbourhood node to which the intermediate data transmission must be sent must offer a balance between hop count and connection quality. The next node should be close to the first node, have a wide coverage area around it, and, with relation to other nodes, be on the anticipated path to the target. In contrast to a single long-distance transmission that would overload the receiving sink node and use even more energy in the transmitting node, the loss of data packets would be handled by successive intermediate transmissions along the intended path if certain cooperation principles were followed. Therefore, cooperative wireless sensor network communication outperforms alternative means of transmission in terms of energy efficiency and data loss-free transmission.

4.2 Algorithms for Machine Learning

Machine learning is a method for making predictions about the future by using historical data. In order to solve classification and clustering problems, supervised, unsupervised, and semi-supervised methods are primarily employed in machine learning (ML) techniques. The value of root depth is needed in this proposed system in order to do the water content prediction. As a result, we first created a prediction model to determine the root depth, and the second prediction model will use the first model's anticipated value to determine the amount of water that is needed.

Since the root depth is a continuous variable, a machine learning approach like regression models that can handle problems involving continuous values must be used. Here, a linear regression model is employed since it is the most effective at forecasting one dependent variable (root depth) over a number of independent factors (time period, crop attributes, and climate conditions). The assumption behind linear regression is that the relationship between the independent variables and the dependent variable is roughly linear. Following is a description of this linear relationship:

X is an independent variable, and Y is a dependent variable that depends on X roughly linearly, where a0 and a1 are unknown constants that indicate the slope and intercept of that slope in a linear model, also known as model coefficients. Here, x stands for days and y for root depth. The idea that root depth is regressing onto days is then advanced. It is provided as follows:

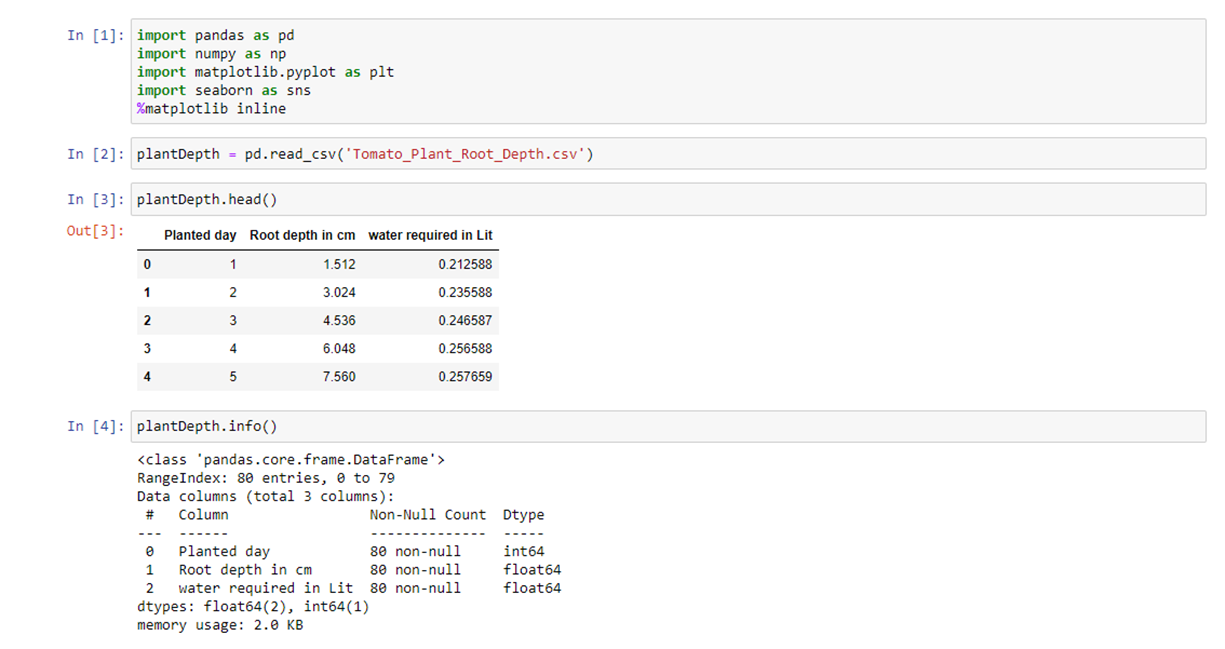
After the model has been trained, we may forecast future root depth values based on a specific number of days that have elapsed since the prediction model was developed by using the coefficients a0 and a1 to obtain the values a 0 and a1. The formula for this is y=a0+a1 x.

Fig 3: Algorithm Description

As illustrated in Fig.3, the proposed framework is assessed using the linear regression algorithm in a Jupiter notebook environment. The necessary packages, including Pandas, NumPy, Matplotlib, and seaborn, are first imported. The Pandas package makes it considerably simpler to import, manipulate, clean, and analyse numerical data. The main Python package for scientific computing that offers multidimensional array object manipulations is the NumPy library. A wide variety of data visualisation techniques are available thanks to the Matplotlib software, which also offers an object-oriented API for using plots like graphs with multipurpose GUI toolkits. Seaborn is a matplotlib addon that offers advanced charting methods that work with data frames, arrays, and entire datasets. To visualise the related outputs within the execution context, the Matplotlib library is made inline as shown.

One compatible dataset, known as Tomato plant root depth in csv format, is created using data cleaning techniques using the information gathered in module one. It shows the number of planted days, the root depth in centimetres, and the necessary water content in litres. The number of days since the crop was planted in a farm field is represented by the 64-bit integer datatype known as the "planted day" column. The root depth column represents the length of the root as it has grown over time. The last column represents the water content needed for one crop with the set root depth on that specific day and is represented by float data type of 64 bits. It is of 64 bits.

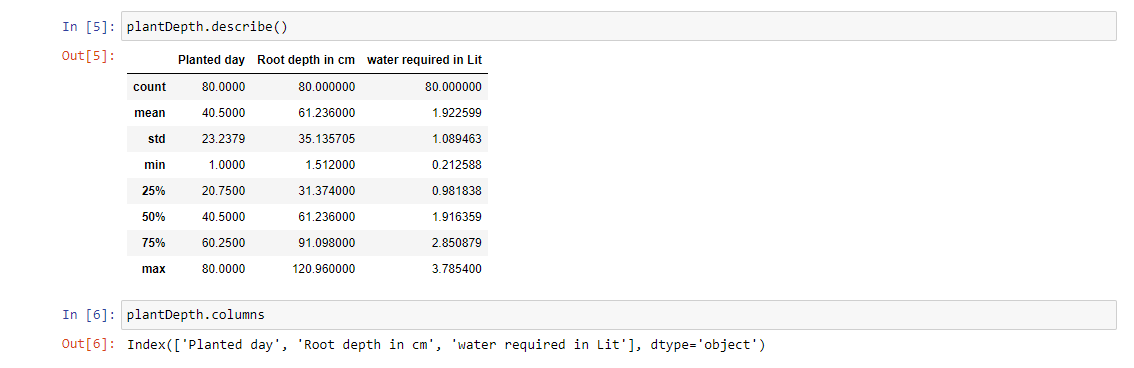


Fig 4: Dataset Range Description

For computing statistical data like the number of rows, the mean value of each column, the standard deviation, the minimum and maximum value that is possible, etc. in Python, use the describe () function. As a result, using the describe function will give us a quick overview of the full dataset. The regression procedure may now be used to create the prediction model after we have an overview of the full dataset and the value distribution among the variables. The machine learning technique is used in a sequence of phases to construct the prediction model. The array that will hold the column values of the independent (x) and dependent (y) variables is given the independent (x) and dependent (y) variables. Once the variables are decided, the dataset will be divided. Numerous built-in features of the scikit-learn package make this procedure easier.

The built-in function train\_test\_split() is one of such. The dataset will be divided into train and test datasets by this existing function. The split happens at the predetermined size, which is often a train-to-test ratio of 60% to 40% or 70% to 30%. This ratio is used as the test\_size argument.

Fig 5: Fitting data and Training set in prediction model

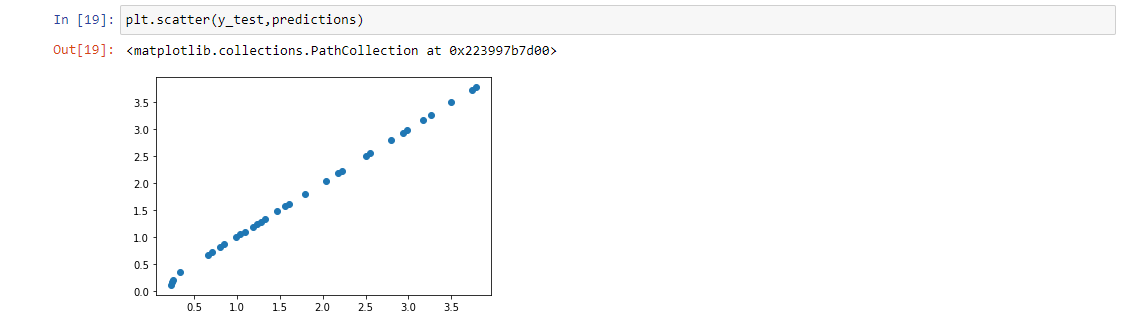
****In the final graphing, a scatter plot technique is employed to observe the relationship between the variables, and the relationship between the variables is represented by dots. The method scatter() in the matplotlib library is used to construct a dispersed dots diagram along a linear line to carry out the scatter plot technique. The more mismatched relationships between the variables are shared, the farther the dots are shown. Scatter plots are frequently used to show the link between various variables and how changing one affects another. For the purpose of evaluating the accuracy of the prediction model, these effects and relationships are carefully examined by looking at the distribution of plots as indicated above.

Fig 6: Prediction Model

In Fig. 6, the final prediction is produced for the fresh dataset that has just arrived, and the scatter plot technique stated above is used to visualise the results. The scatter plot seems to be very thin in the final results since all of its dots were displayed along the linear line and are parallel to one another. This indicates that there is no appreciable difference between the predicted values and the values of the trained dataset. As a result, the projected values match those that the prediction model generated when it was trained using the training dataset. As a result, the model makes the expected predictions about the new incoming values. Additionally, its effectiveness will be covered in the result section.

4.2.1 Estimation of Water Need

The above-mentioned estimate about the water content only applies to a single plant seedling. As a result, it is necessary to estimate the amount of water needed to irrigate the entire farmland with "n" number of saplings. Plants need water to flow continuously, such as paddy, etc. The method used to estimate the amount of water needed will entirely depend on the crop we are choosing. According to measurements, a fully developed tomato plant needs a space that is around 21 inches wide. i.e., the area of 1.75 foot. Lands, which is measured in acres. A square acre measures 43560 ft. The farmers would be able to plant 14,233 saplings on an acre of farmland. Equation (6) can be used to determine how many saplings can be planted on an acre of farmland.

The quantity of plants to be sown is therefore estimated. At their full maturity, all of these plants will need 1 gallon of water at the height of the sun. This is the volume of water needed for one mature plant to grow in well-drained soil during the hottest part of the day. An acre of tomato plants, or 14,223 of them, will need a water content of approximately 55 litters, which is rounded up to 53,840 litters of water each day at the end of harvesting in the worst climate conditions. The water content required is 1.2 inches at an average time, which equates to 2.36 litters of water per day per sapling. There are 980 barrels total, each holding 55 gallons. Although it may appear like a lot, the real amount of water utilised in conventional techniques of irrigation is far smaller when compared to the scale of mass production of agricultural products (in this case, tomatoes). Due to its higher evaporation avoidance, it is estimated that a maximum of 40–60% less water is needed overall compared to older approaches.

This estimated number is meant to demonstrate the water need estimation for a given acre of agricultural land during a specific period close to harvesting. This value will not remain constant over the course of farming, which is one of the reasons for employing ML.

4.2.2 Streamlit API-based Water Estimation and Display

A Python tool called the Streamlit API enables us to build web apps for machine learning. This Streamlit software serves as the backend for real-time water estimation in this case, and it displays the results on a. The machine learning algorithm is implemented in a Streamlit API-based web app as shown in This is the same algorithm that was explained in the machine learning implement part, but in this case, Streamlit is used to execute it and display the results for the user. The Jupiter notebook Python code is located on a page called notebook.py, and implementation begins by importing all necessary libraries, metrics, and functions. Here, an internal function called function\_pred() contains the algorithm. The last line of the function\_pred() method's function\_pred() method uses the return statement to return the value that the predict() function computed and stored in a variable called predictions.

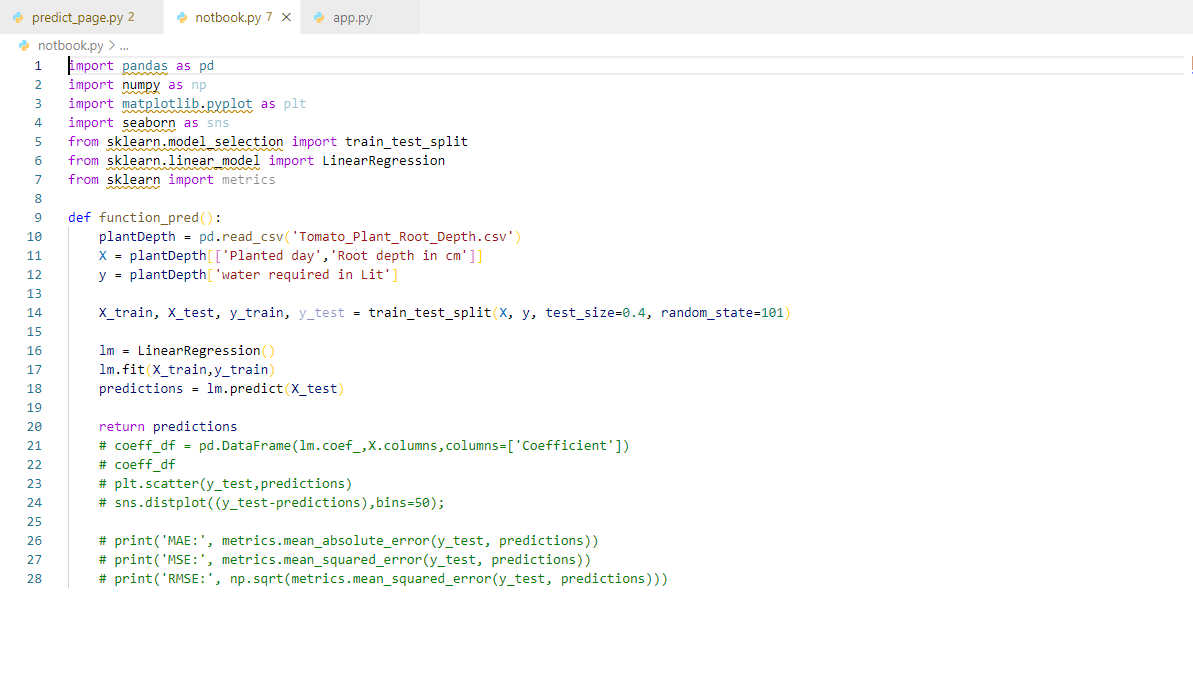
****

Fig 7: ML implemented in Streamlit API

Predict\_page.py is the name of the webpage in this instance since it is designed to show the output of a machine learning prediction model after calling a function and estimating the total amount of water needed for the required amount of farming space. It begins by importing all required libraries for streamlit, including no\_type\_check, streamlit, pickle, etc., and then the notebook. This import of the Python page allows access to the function\_pred(). The entire coding snippet is written inside a function named show\_pred\_page() after all libraries have been imported. The Streamlit elements like St.text\_input, St.date\_input, St.number\_input, etc. are used in the show\_pred\_page() method body, which also contains code. The user can enter the information needed for the computing process in a text box provided by these components.

The webpage's heading reads "Water content prediction for best irrigation practise" and invites users to provide the following information to determine their water needs. The user can then select a crop they have planted for farming, such as a tomato, potato, brinjal or maize, using a drop-down box after that. Following that, a date picker prompts the user to choose the day when crops are planted. The following calculation subtracts this selected date from the automatically selected current date to find the number of days between the current date and the plantation date. The users are then asked to enter the total number of acres used for farming.

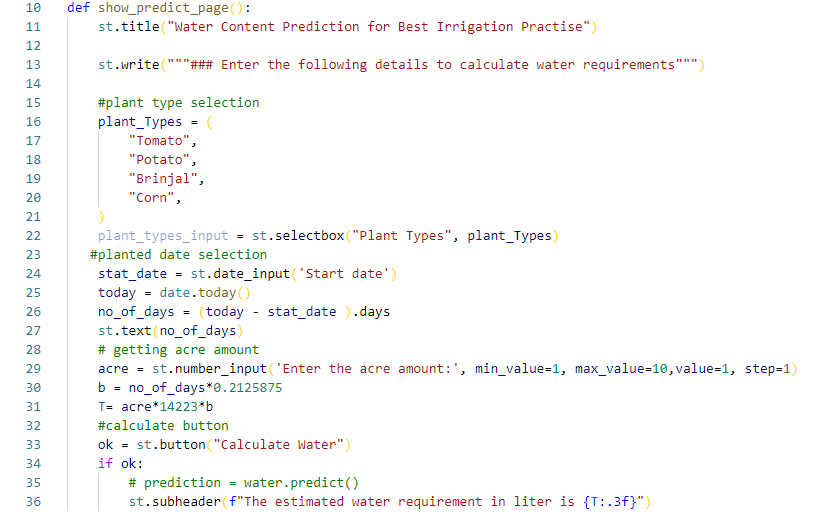
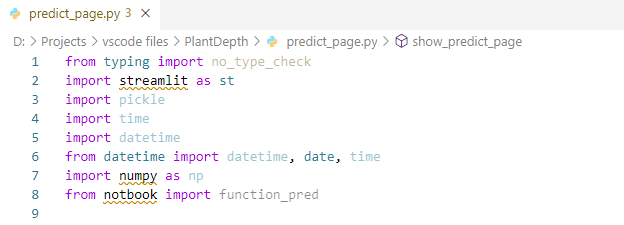
****

Fig 8: Streamlit based backend of web application for result display

After collecting all of these data from farmers, the root depth is calculated. The machine learning prediction model predicts that the root depth will increase linearly, thus a coefficient value that represents how much, on average, the root depth increases each day is determined to be a constant value. It grows at a rate of 1.5112 centimetres per day, which, when spread across three seasons, or 80 days, results in a maximum length of 120 centimetres. In order to calculate the root depth for the current date, the estimated value of the number of days is multiplied by the linear growth rate coefficient constant.

The entire amount of water needed is approximated at the end of the calculation. For this, a linear relationship between root depth and water requirement is established in the ML implementation and water need estimation section, and it is discovered that for a minimum root depth of 1.512 centimetres, approximately 0.212588 litres of water, or 212 millilitres per day per plant, are needed for tomato planting. And in order to increase the value of water demand based on increasing the value of root depth, a linear calculation of root depth vs. water requirement is used. In order to determine the amount of water needed for one acre, the 'b' value from the previous calculation (for one plant) is multiplied by 14223, or the total number of plants that might be planted in an acre (calculated from section 4.2.1). And finally, the final value of water need for the total number of farming acres, which is given by 'T' in the function show\_predict\_page(), is obtained by multiplying the last input of the number of acres by this intermediate value.

A function call will be made from a Streamlit page called app.py for the show\_predict\_page() function when the user clicks the button to calculate water and the Boolean value is true. This function call will cause the predicted water requirement to be displayed in litters on the web page. The results and comments section displays the output that was produced. The actual implementation of the proposed system does not include API display; instead, all computation and implementation will take place within IoT devices, and decisions regarding the irrigation process are made in the IoT automation circuit itself. This web application was created to display the water need estimation portion of the system.

4.3 Setting Up a Root Zone Injection

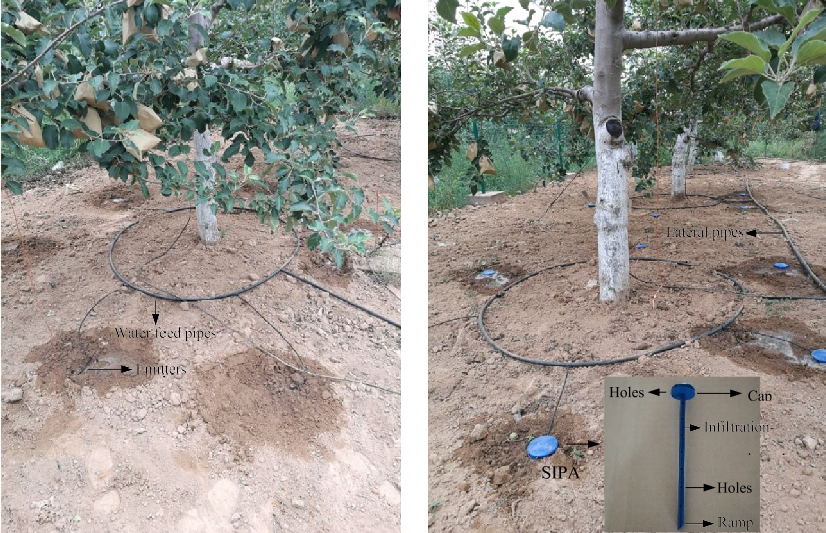
 Overusing water resources will stress plants, causing them to deteriorate faster than is necessary. Similar to overstressing, not using as much water as is necessary will result in problems with plants drying out and dying. Utilising the injection method of water conduction, which is directed aimed to deliver at the root of each and every plant as illustrated in Fig. 9, after carrying out the water estimation process that is described in the ML implementation section, can eliminate the issues of overstressing and under-usage of water resources.

Fig 9: RII method of irrigation

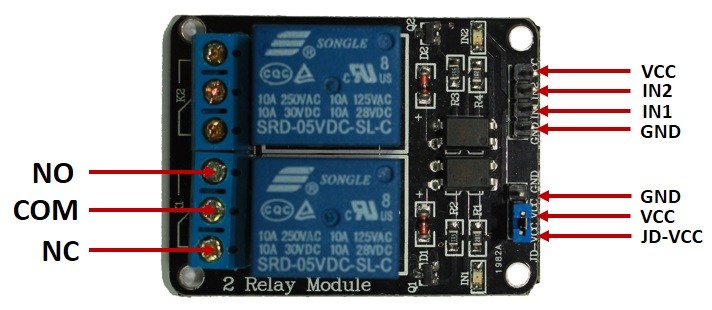
It explains how the pipes are linked so that water resources can be delivered directly to the plant roots. An automated irrigation system configuration that uses a microcontroller to determine the time period of water delivery and the time interval gap to shut down the system before the next irrigation process to start can be used to control this mechanism. The entire system can be designed to deliver water resources in accordance with farming needs. To prevent plants from dying from overheating, overcooling, and overstressing problems that develop in each of the many climatic situations, watering should be done regularly twice a week in the autumn, three times a month during rainy seasons, and once per day during high temperature sunny seasons. The proper time interval of irrigation utilising the root zone injection method is planned and implemented using IoT hardware systems in real-time agricultural fields based on all of these users, climatic, and environment criteria. The following section of IoT implementation goes into further information about this IoT hardware solution.

4.4 IOT Implementation

The suggested system's final module utilises machine learning algorithm outputs to implement autonomous precision irrigation over the Internet of Things. This module automates irrigation by employing IoT devices like an Arduino Uno board, a water pump, a relay, and fuzzy logic commands to regulate the on and off states of water pump valves. The open-source Arduino microcontroller features 14 I/O pins and an ICSP connector for in-circuit serial programming. Based on the programming that is contained inside, this microcontroller allows us to carry out specific tasks (in this case, operating water pump valves). In Fig.10.1, the Arduino Uno board is described.

Fig 10.2: Relay Module

Fig 10.1: Arduino Uno Board



The Arduino activates the relay module as soon as it issues the command to start the irrigation process along with an estimation of the water content. This will control how valves that control irrigation are turned on and off. Since the sophisticated relay module can operate with both AC and DC current and performs better than conventional transistor circuits, it guarantees an effective control system for automated water distribution. The relay module that regulates the pump's active time is then placed together with the water conducting pump. The neutron probes that were bored into the soil layer will be attached with RII injection nozzles that receive water from the water pump. Water is supplied to underground root zones where roots are densely populated.

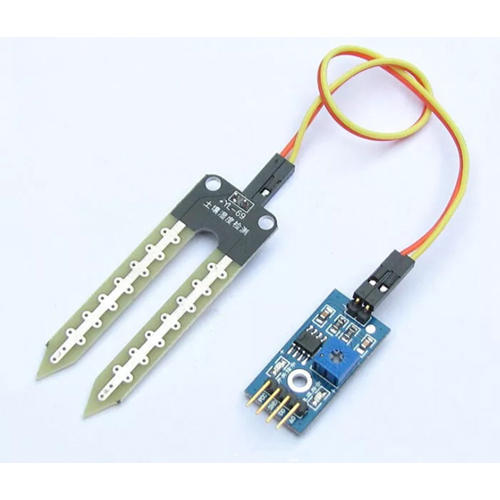
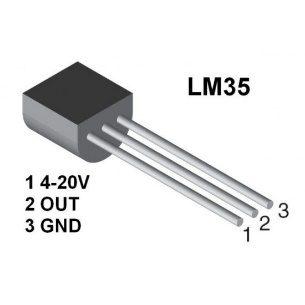


Fig 10.3: Water pump

Fig 10.5: Moisture Sensor

Fig 10.4: Temperature sensor

In the first module, where the field data were gathered, a temperature sensor and a moisture sensor are utilised. These sensors are shown in Figs. 10.4 and 10.5 respectively. However, they might also be applied in the system's final module, which would execute adaptive watering based on factors like rainfall and sunny time zones [11]. The complete IoT setup, which needs to be programmed with fuzzy logic control commands for automated water valve management in the precision irrigation, will be ready for deployment once all of the above-mentioned IoT devices and hardware components have been put together. Figure 11 depicts the finished IoT configuration.

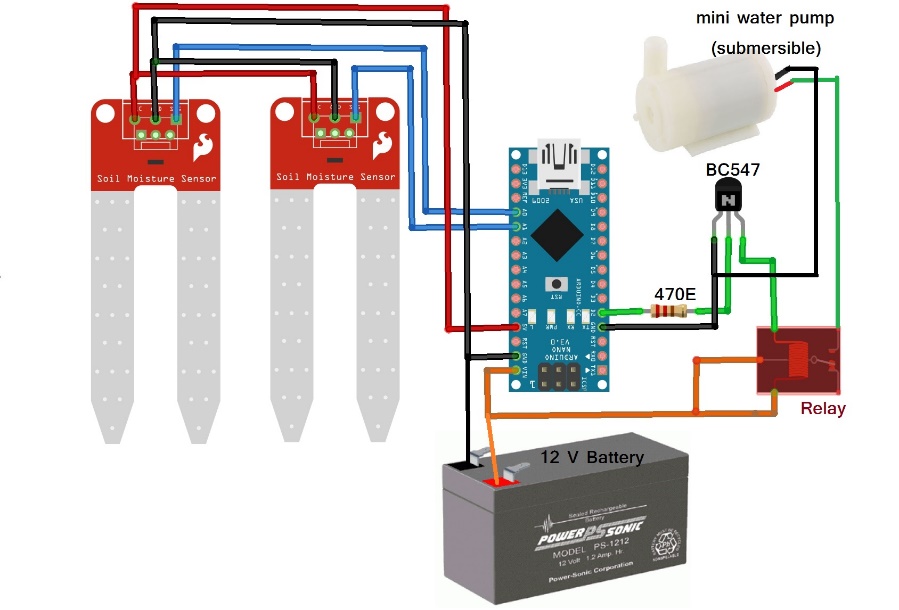


Fig 11: IOT setup for automated Irrigation Driven by fuzzy logic control command.

The detailed linkages between the IoT components are described in the Fig. 11 IoT model for automated precision irrigation. Here, the sensors were linked to an Arduino board to convey information about the farming environment. Since Uno is the microprocessor, the sensors, relay module, battery, LED indicator, and water pump were all connected to Arduino for control operations. Real-time implementation will replace the setup's battery with a power source, and the water pump's capacity will be scaled to fit the needs. As the practise develops over the course of farming time since it is heuristic and subjective, the LED indication will let us know if the pump is turned on or off, and the activation commands will be passed by fuzzy logic with absolute precision.

This completes the proposed system and includes every step of the implementation process, from module one of wireless sensor networks through module three of IoT implementation of precision farming.

1. Results and Discussions.

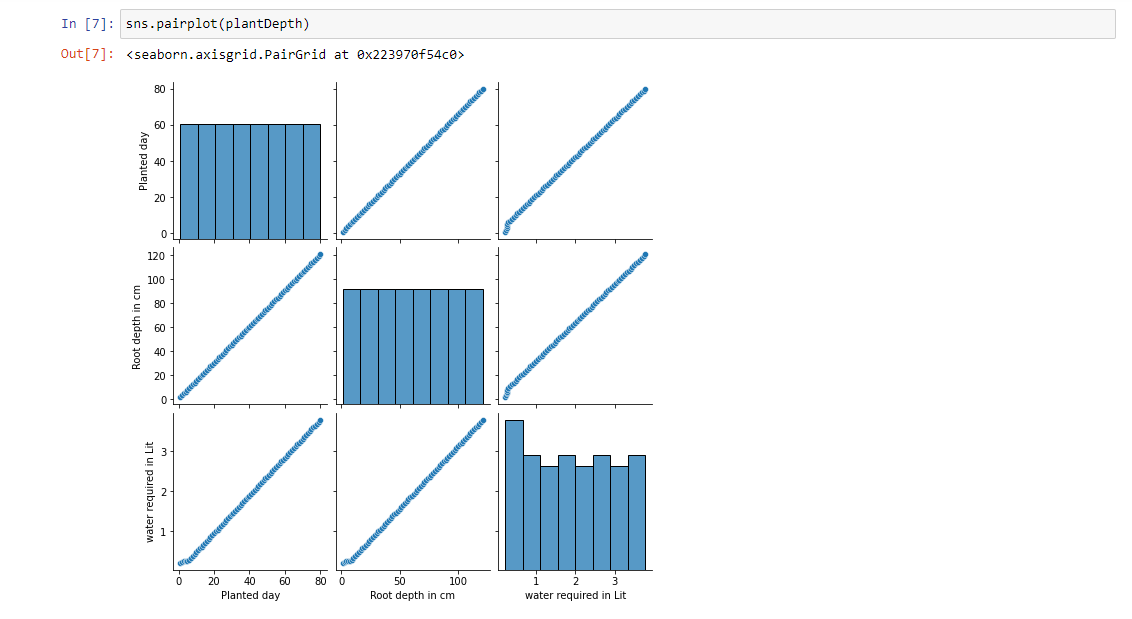
****The results of putting the proposed system into practise are assessed and thoroughly discussed in this section in order to draw out its superior advantages over the current systems of the conventional SDI method and other automation-driven approaches to irrigation. The findings are assessed using the data analytics tools in the Jupiter Notebook, and the efficiency and accuracy of the suggested system are estimated using coefficient metrics from the Sci-Kit Learn package, including error ratios, MAE, MSE, and RMSE. As a result, the machine learning method to data analytics is utilised to assess the system's overall results.

Fig 12: Pair-plot Comparisons

As shown in Fig.12, the pairplot of each column's values vs each other column's values is rendered using the seaborn library. As the number of planted days increases, it can be seen that the value of root depth rises linearly. Additionally, the necessary water content vs. root depth plot shows the similar trend of linear escalation. Thus, it is abundantly evident that as the number of days increases, root depth grows, and that the amount of water needed to maintain that depth likewise grows. According to the association rule, the relationship between the planted day and the amount of water needed also tends to increase linearly.

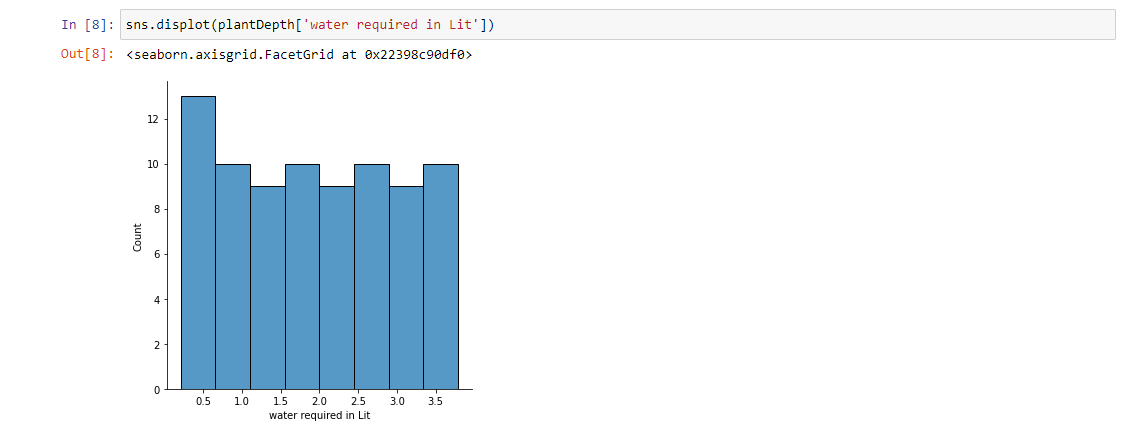
Plotting identical columns, such planted days against planted days or root depth against root depth, reveals a steady slope because the recorded values for the two compared columns were the same. However, the similar water required vs. water required column displays a variable number since the amount of water required varies daily or weekly depending on a variety of factors like the amount of sunlight, the amount of rainfall, and the amount of retained ****moisture.

Fig 13: Displot representation of water Requirements

In Fig.13, the estimated quantity of water is shown over a range of days to represent the required amount in litters. We can better understand the variation in data distribution by using a distribution plot or displot. Here, the x-axis represents the amount of water needed, and the y-axis represents the number of days. The graph demonstrates that during the first 13 days of plantation, 0.5 lit of water is needed because the plant was just beginning to bloom. Then, for the following 10 days, or 14 to 24 days of plantation, a litter of water is needed. The sequence of water needed for a certain time period is shown in Fig. 13 throughout the complete farming season, which is 80 days (about 3 months).

This three-month farming cycle is divided into three distinct phases based on the degree of plant maturity: early crop development, midseason, and late season. Based on their botanical qualities, the amount of water required will vary for each of these maturity seasons. As a result, the output of the plot representation is an estimation of the water that changes over time in response to the weather. Furthermore, by increasing the number of histogram bins to provide more distinct values, it is possible to see the correctness of a more complete illustration of this representation.

The heatmap technique will assist us in visualising the intensity of values scattered across all three columns for an even better description. It was shown using a color-changing heat-map scale and spans from the smallest to the maximum gradian values. This min-max gradian, which depicts the value distribution by density in Fig.14, ranges from 0.9 to 1.0 with all minor variations in between.

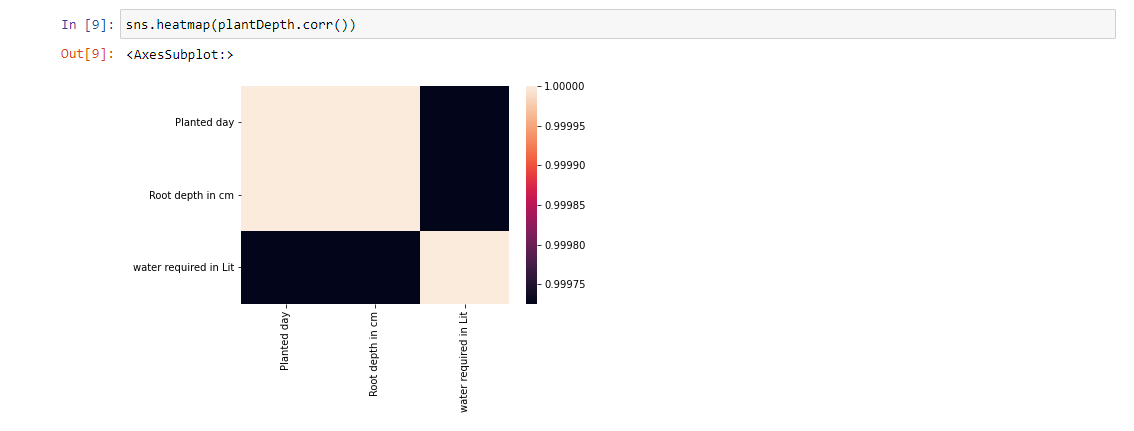
****

Fig 14: Heat-map illustration

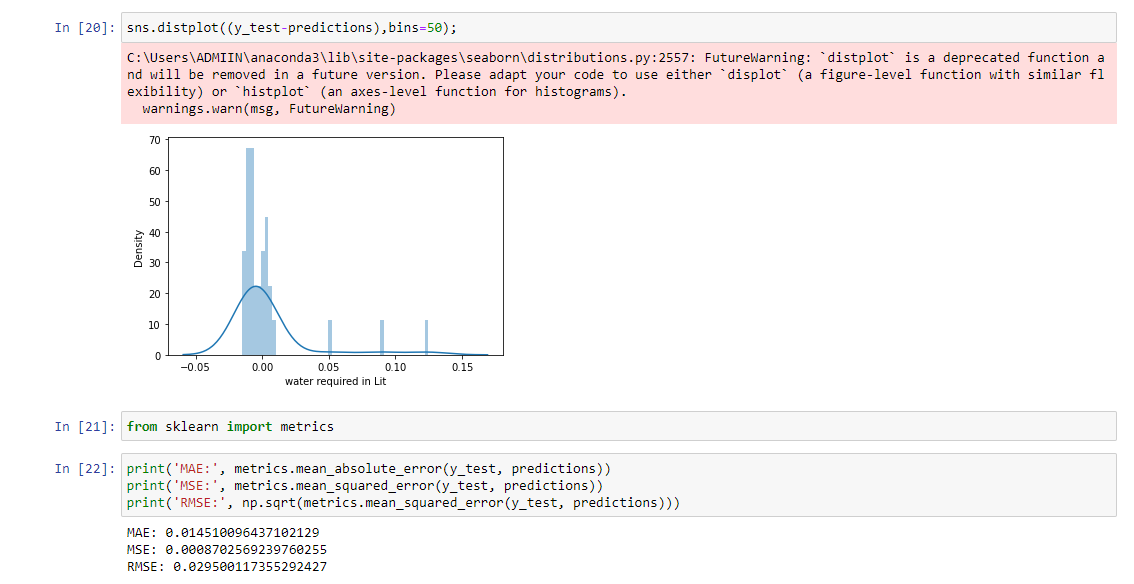
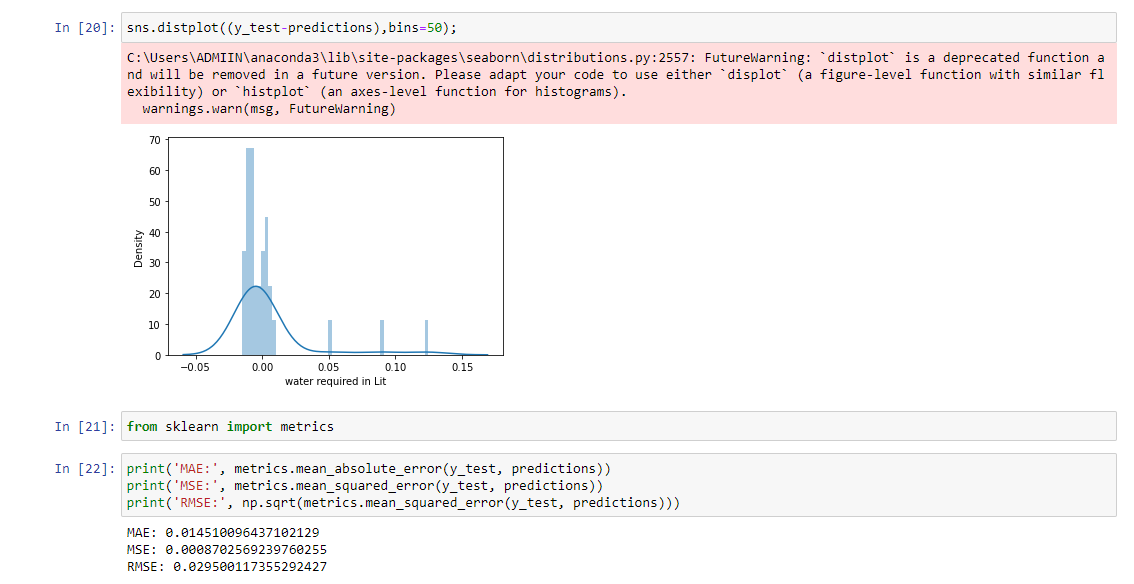
****Similar to the pairplot comparison, constant values between related columns also exist here with a maximum gradient of 1 because they are constant values, and linearly fluctuating incremental values between different columns were distributed in the range of 0.975 to 0.980 in the heat map scale. Our understanding of where the data gathered from farms is substantially concentrated and where it is slightly distributed is improved as a result. The areas that need to be irrigated with water are thus identified based on the heat-map distribution of the data and output values and the suitable water estimation.

Fig 15: Metrics Evaluation for model accuracy

Fig.15 shows the results of the final assessment of the prediction model's accuracy. Here, measurements from Scikit Learn are used to generate the model's error percentage. Metrics, such as mean absolute error (MAE), mean squared error (MSE), root mean square error (RMSE), and others, are tools in the Scilit Learn package for evaluating the accuracy of models. These metrics use inputs like the total number of data points, observed and predicted values, and true value to calculate the mean and estimate the highest difference between predicted and original values, which in turn creates the error percentile of prediction model accuracy. The MAE, MSE, and RMSE metrics formulas are shown in the following equations.

Where MAE = mean absolute error, n = number of data points, Yi= prediction value, and xi= true value in the corresponding input dataset.

Where MSE = mean squared error, n = number of data points, Yi= observed values, Y^i= predicted values in the corresponding input dataset.

Where RMSE = root mean squared error, N = number of non-missing data points, i = variable i, xi=actual observed values, x ^i= estimated values. The RMSE can also be said as the squared root value of MSE, thus an advanced method of calculating precision that magnifies small errors that goes unnoticed in other methods, by squaring them and then taking root value of the squared values.

Scikit Learn's library imported the aforementioned measurements into Fig. 15 and performed error rate estimation. And the findings demonstrate that the prediction model's error percentile tends to be 2% in RMSE measurements and 1% in MAE metrics. This appears to be a very accurate prediction model in this experiment, but when applied to real-time agricultural practise, results will vary greatly and error rates will rise due to a variety of volatile data that will be recorded from real-time farming.

The web application created in the Streamlit API section can be utilised to show users the results because the model is demonstrated to be accurate in terms of error rate. The online page with the heading "Water Content Prediction for Best Irrigation Practise" is displayed in Figs. 16.1 and 16.2. This website includes a "Calculate Water" submit button and a user interface for capturing three user inputs.

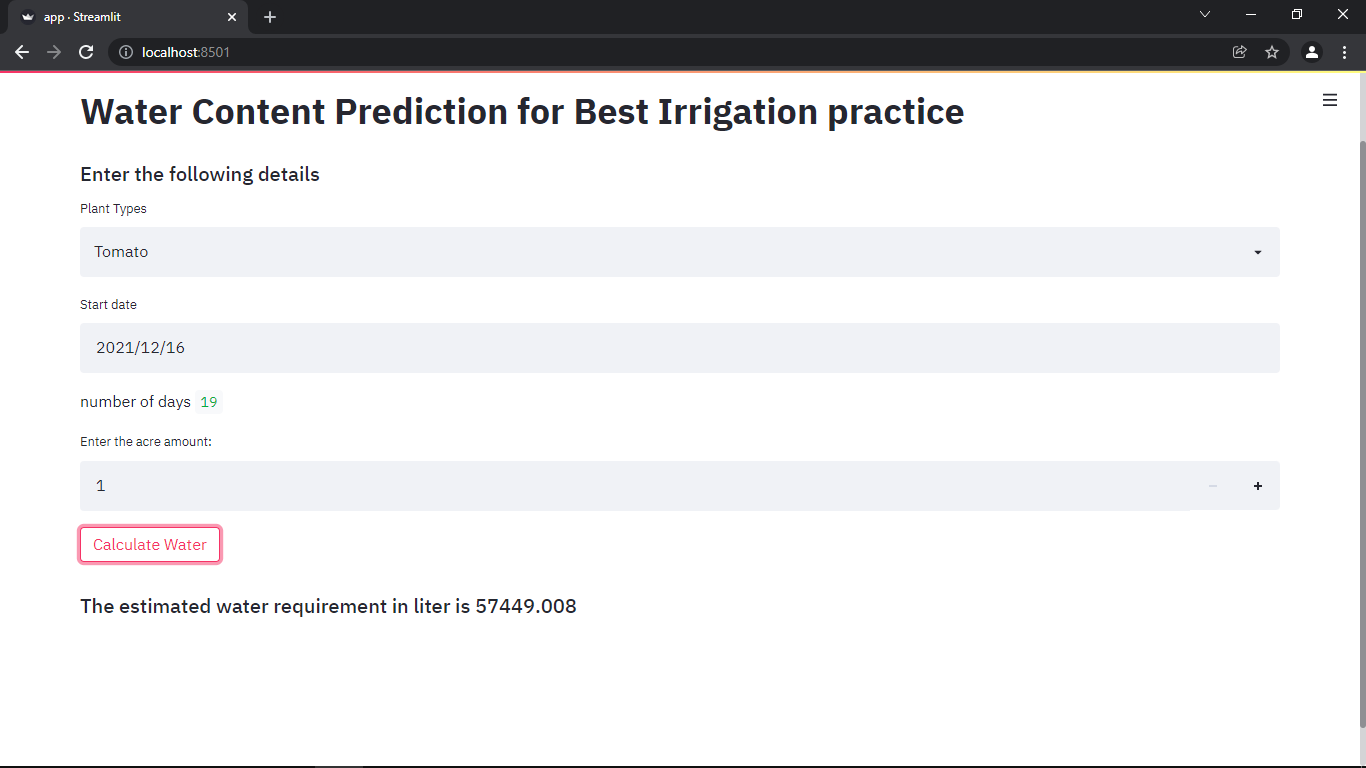


Fig 16.1: Result display through web application for 1 acre

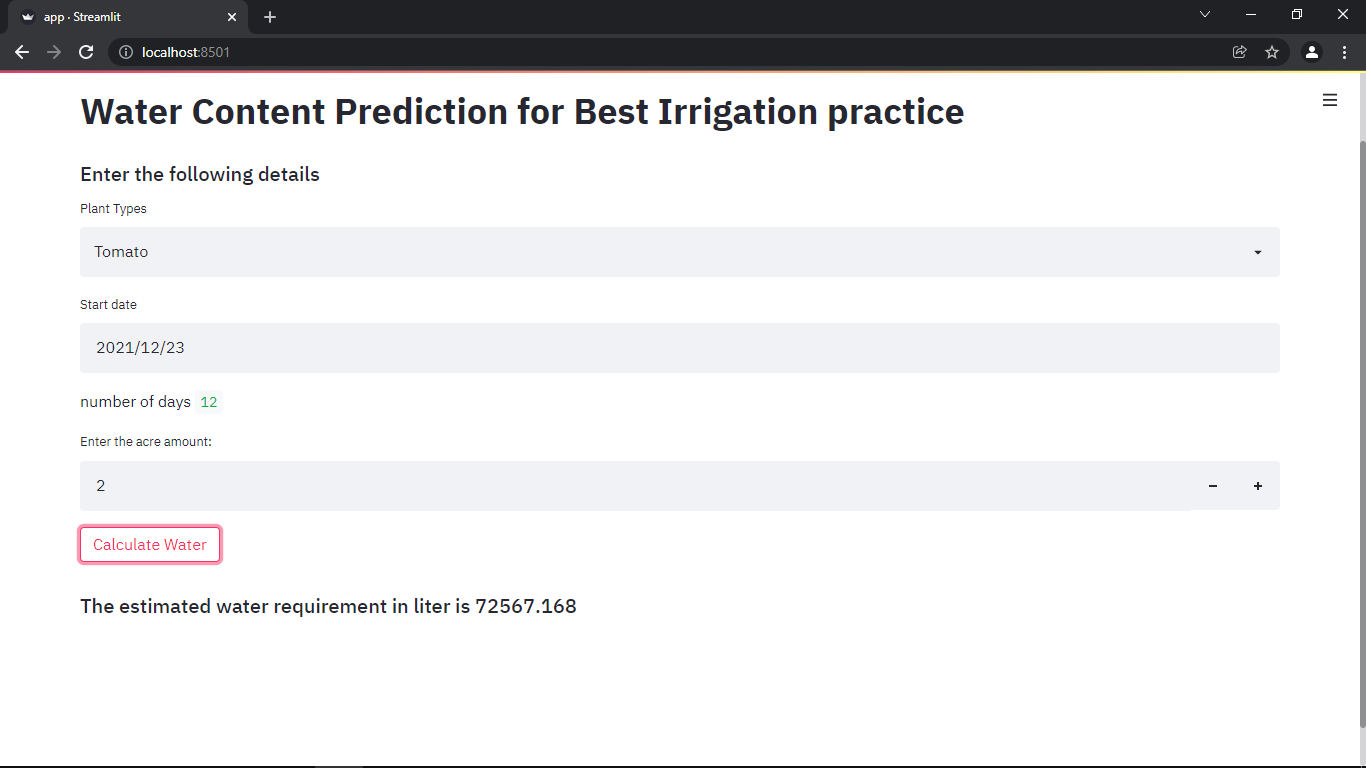


Fig 16.2: result display through web application for 2 acres

Three inputs must be entered by the users: the type of plant that was planted for farming, such as tomato or sugarcane; the day that the plantation was completed, or the first day of agriculture (this will act as the initial growth phase from which the necessary time period calculations will be made); and the quantity of acres that the user cultivated the specified plant species on. The ML method will import a specific model made for that crop type from the first input. distinct root systems, such as tap roots and fibrous roots, maturation phases, such as s-growth types and linear growth types, distinct water need cycles, and whole diverse botanical traits for plant metabolism and nutrient intakes are present in each variety of crop.

The root depth and water requirements for the given plant type will therefore be determined by importing a plant-specific prediction model. The system will determine the time span between the point of planting and the present date by automatically selecting the current date and subtracting the difference from the second input, or the day on which the plantation had been completed. The date of plantation is taken into account as the start of the maturity phase for all time period analyses and computations in the prediction model. The third input is an integer type representing the number of acres that might be used for planting plants as well as the total amount of water required for that farmland.

As can be seen, two alternative planting dates have been made, and the expected period between them is 19 days and 12 days for mature plants. As was already mentioned in the implementation section for the Streamlit API, this value, along with the previous input, will be fed into the logic that estimates the amount of water that will be needed to irrigate the specified plant species—in this case, a tomato crop—for the specified number of aces on a given day. The end results are thus shown in numbers as 57449.008 litters for the cultivation land with a 19-day crop maturity of one acre and 72567.168 litters for the cultivation field with a 12-day crop maturity of two acres, respectively.

1. Conclusion:

Agriculture needs a significant attention from multiple areas in order for better utilization of its limited resources that is getting scarce every year. Precision irrigation and water-conservation practises are especially important in dry locations. In order to predict the root depth of cultivated crops, we first developed a module for data collection and energy-efficient transmission via wireless sensor networks. Based on the predicted root depth and processed weather data, an automated precision irrigation practise is then carried out with the aid of IoT based automated irrigation setup. The root zone injection way of water delivery is then used to complete the overall irrigation.

Thus, reducing extra usage of water, and preventing evaporation wastage. The effectiveness as mentioned in the results section is significantly higher than both the current automated irrigation techniques used in farming and all traditional approaches. However, while using the suggested method in real-time agricultural practise for a large area with highly dynamic data obtained, the accuracy and efficiency may be impacted and reduced. Thus, it will soon be necessary to integrate more ML algorithms as well as botanical traits of planted crops such bark width, nutrient needs for plant metabolism, location, and soil-based trait alterations. Future development work will also take into account and implement scalability of the system, secured implementation to prevent intrusions, and supporting numerous crop varieties individually with particular ways.

References:

1. S.R. Barkunan, V. Bhanumathi, J. Sethuram, “Smart sensor for automatic drip irrigation system for paddy cultivation” Received 30 January 2017, Revised 13 June 2018, Accepted 13 November 2018.
2. Avşar, E., Buluş, K., Saridaş, M.A. and Kapur, B., 2018, May. Development of a cloud-based automatic irrigation system: A case study on strawberry cultivation. In *2018 7th International Conference on Modern Circuits and Systems Technologies (MOCAST)* (pp. 1-4). IEEE.
3. Kodali, R.K. and Sarjerao, B.S., 2017, July. A low-cost smart irrigation system using MQTT protocol. In *2017 IEEE Region 10 Symposium (TENSYMP)* (pp. 1-5). IEEE.
4. Jaafar, M.F.M., Hussin, H., Rosman, R., Kheng, T.Y. and Hussin, M.J.A., 2019, October. Smart Cocoa Nursery Monitoring System Using IRT for Automatic Drip Irrigation. In 2019 IEEE 13th International Conference on Telecommunication Systems, Services, *and Applications (TSSA)* (pp. 108-113). IEEE.
5. H. JOCHEN SCHENK and ROBERT B. JACKSON, “Blackwell Science, Ltd Rooting depths, lateral root spreads and below-ground/ above-ground allometries of plants in water-limited ecosystems” Department of Biology and Nicholas School of the Environment and Earth Sciences, Duke University, Durham, North Carolina 27708, USA.
6. Yan-Ping Wang, Lin-Sen Zhang, Yan Mu, Wei-Hong Liu, Fu-Xing Guo1 And Tian-Ran Chang, “Effect of a Root-Zone Injection Irrigation method on water productivity and Apple production in a semi-arid region in north-western china” DOI: 10.1002.
7. Chiyurl Yoon, Miyoung Huh, Shin-Gak Kang, Juyoung Park, Changkyu Lee, “Implement Smart Farm with IoT Technology” International Conference on Advanced Communications Technology (ICACT).
8. K. A. Patil, N. R. Kale, “A Model for Smart Agriculture Using IoT”, 2016 International Conference on Global Trends in Signal Processing, Information Computing and Communication.
9. Rajinder Kumar, Nagaraj V Dharwadkar, “A Wireless Sensor Network Based Low Cost and Energy Efficient Frame Work for Precision Agriculture”, 2017 International Conference on Nascent Technologies in the Engineering Field (ICNTE-2017).
10. Harmantoa, V.M. Salokhea, M.S. Babelb, H.J. Tantauc, “Water requirement of drip irrigated tomatoes grown in greenhouse in tropical environment”,71 (2005) 225–242.
11. Bright Keswani, Ambarish G. Mohapatra, Amarjeet Mohanty, Ashish Khanna, Joel J. P. C. Rodrigues, Deepak Gupta, Victor Hugo C. de Albuquerque, “Adapting weather conditions based IoT enabled smart irrigation technique in precision agriculture mechanisms”, The Natural Computing Applications Forum 2018.
12. Weiwei Fang, Feng Liu, Fangnan Yang, Lei Shu, Nishio, S. (2010), “Energy-efficient cooperative communication for data transmission in wireless sensor networks”, 56(4), 0–2192. doi:10.1109/tce.2010.5681089.
13. Huang Lu, Jie Li, Guizani, Mohsen (2014),” Secure and Efficient Data Transmission for Cluster-Based Wireless Sensor Networks”, IEEE Transactions on Parallel and Distributed Systems, 25(3), 750–761. doi:10.1109.
14. M. Nesa Sudha; M.L. Valarmathi; Anni Susan Babu (2011), “Energy efficient data transmission in automatic irrigation system using wireless sensor networks”, 78(2), 215–221. doi:10.1016.
15. Hsu, T.-H.; Yen, P.-Y. (2011),” Adaptive time division multiple access-based medium access control protocol for energy conserving and data transmission in wireless sensor networks”, 5(18), 1–. doi:10.1049/iet-com.2011.0088.
16. Liu, Xuesong; Wu, Jie (2019), “A Method for Energy Balance and Data Transmission Optimal Routing in Wireless Sensor Networks. Sensors”, 19(13), 3017–. doi:10.3390/s19133017.
17. Chikankar, Pravina B.; Mehetre, Deepak; Das, Soumitra (2015), International Conference on Pervasive Computing (ICPC), “An automatic irrigation system using ZigBee in wireless sensor network”, doi:10.1109/PERVASIVE.2015.7086997.
18. Benaddy, M.; Habil, B. El; Ouali, M. El; Meslouhi, O. El; Krit, S. (2017), International Conference on Engineering & MIS (ICEMIS), “A mutlipath routing algorithm for wireless sensor networks under distance and energy consumption constraints for reliable data transmission”, doi:10.1109/ICEMIS.2017.8273076.
19. Ma, Ruiping; Xing, Liudong; Michel, Howard E. (2007), “A New Mechanism for Achieving Secure and Reliable Data Transmission in Wireless Sensor Networks”, 274–279. doi:10.1109/THS.2007.370058.
20. Kaewmard, Nattapol, Saiyod, Saiyan (2014). [IEEE 2014 IEEE Conference on Wireless Sensors (ICWiSe) - Subang, Selangor, Malaysia (2014.10.26-2014.10.28)] 2014 IEEE Conference on Wireless Sensors (ICWiSE), “Sensor data collection and irrigation control on vegetable crop using smart phone and wireless sensor networks for smart farm”, doi:10.1109/icwise.2014.7042670.
21. Dr. S. Velmurugan, V. Balaji, T. Manoj Bharathi, K. Saravanan, “An IOT based Smart Irrigation System using Soil Moisture and Weather Prediction”, International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181.
22. George Kokkoni, Sotirios Kontogiannis, Dimitrios Tomtsis,” A Smart IoT Fuzzy Irrigation System”, IOSR Journal of Engineering (IOSRJEN), ISSN (e): 2250-3021, ISSN (p): 2278-8719.
23. Aashika Premkumar, Thenmozhi K, P Monishaa, Padmapriya Praveenkumar, “IoT Assisted Automatic Irrigation System using Wireless Sensor Nodes”, 2018 International Conference on Computer Communication and Informatics (ICCCI -2018).
24. Revanth Kondaveti, Akash Reddy, Supreet Palabtla, “Smart Irrigation System Using Machine Learning and IOT”, doi:10.1109/ViTECoN.2019.8899433.
25. A. Mahesh Reddy, K. Raghava Rao, “An Android based Automatic Irrigation System using a WSN and GPRS Module”, Indian Journal of Science and Technology, Vol 9(29), DOI: 10.17485, August 2016.
26. Shiraz Pasha B.R., Dr. B Yogesha, “Microcontroller Based Automated Irrigation System”, The International Journal of Engineering and Science (IJES), ISSN (e): 2319 – 1813 ISSN (p): 2319 – 1805.
27. Getie Dereje Derib, “Cooperative Automatic Irrigation System using Arduino”, International Journal of Science and Research (IJSR), ISSN: 2319-7064.
28. John R. Dela Cruz, Renann G. Baldovino, Argel A. Bandala, Elmer P. Dadios, “Water Usage Optimization of Smart Farm Automated Irrigation System Using Artificial Neural Network”, 2017 Fifth International Conference on Information and Communication Technology (ICoICT).