A Smart Irrigation and Monitoring System

Dr. V Asha¹, Dipsikha Das²
¹,²Department of Master of Computer Applications, New Horizon College of Engineering.

Abstract: Internet of Things, commonly known as IOT is a promising area in technology that is growing day by day. It is a concept where by devices connect with each other or to living things. Internet of Things has shown its great benefits in today’s life. Agriculture is one amongst the sectors which contributes a lot to the economy of Mauritius and to get quality products, proper irrigation has to be performed. Hence proper water management is a must because Mauritius is a tropical island that has gone through a water crisis in either past few years. With the concept of Internet of Things and the power of the cloud, it is possible to use low cost devices to monitor and be informed about the status of an agricultural area in real time. Thus, this paper provides the design and implementation of a Smart Irrigation and Monitoring System which make use of Microsoft Azure machine learning to process data received from sensors in the farm and weather forecasting data to better inform the farmer the appropriate moment to start irrigation. The Smart Irrigation and Monitoring System is made up of sensors which collect data such as air humidity, air temperature, and most importantly soil moisture data. These data are used to monitor the air quality and water content of the soil. The raw data are transmitted to the cloud platform, Microsoft Azure cloud platform, and are processed through a machine learning operation which had to be trained beforehand. The farmer is then informed through either a web app or mobile app as to when to irrigate. The Smart Irrigation and Monitoring System proposed in this paper allows the farmer, through both the mobile app and web app to send command to start the irrigation process.

General Terms: Machine Learning, Internet of Things, Smart Irrigation

Keywords: Internet of Things, Smart Irrigation and Monitoring System, Microsoft Azure, Cloud, Machine Learning

I. INTRODUCTION

Mauritius is a country known for its agriculture and sea[1]. In parallel to growing crops, its irrigation needs must also be taken into consideration. Crops require proper irrigation at appropriate time intervals for them to grow healthily. Agriculture is a field where labor is very crucial and is also in high demand. As revealed in a recent study of the Human Resource Development Council (HRDC), the employment rate in the primary sector has been foreseen to decrease nearly from 8.1% to 4.4% by 2015[2]. There as on behind this decline in workforce was because youngsters were not passionate about the agricultural sector and they did not find much opportunities. As a result, farmers who dedicate their times to grow crops in large areas had to spend their whole day outside to ensure that the crops are being grown properly. Farmers rarely had good controls on crops and also suffered from great losses due to unforeseen, and inconvenient weather conditions. Mauritius being a country where air is very unpredictable[3][17], farmers do not have enough resources to closely monitor their fields so as to know when is the appropriate moment to start irrigating their crops. There are situations where farmers irrigate their crops without knowledge of the weather forecast. As are suit, this situation leads to bad water management. Thus, this paper proposes a Smart Irrigation and Monitoring System (SIMS) which uses sensors to collect air humidity, air temperature, and most importantly soil moisture, and transmit the raw data to a cloud platform (Microsoft Azure cloud platform). The data are then processed, using Azure Machine Learning Algorithms (the two-class boosted decision tree algorithm has been selected to train and
process the data). The algorithm also factors in weather forecast data in order to provide an informed decision to This paper is organized as follows: Section 2 provides a literature review in the domain of smart farming particularly towards irrigation along with a critical analysis. Section 3 provides an overview of the proposed architecture.

**Critics:** This paper assumes mainly the physical data from sensor sand then compare them with weather forecast by the use of decision tree model whose results are shown in Table 1.

![Table 1: Results from Decision Tree Model](image)

| Result from Decision Tree | Smooth sensed Data | Decision Data |
|---------------------------|--------------------|--------------|
| “no rain”                 | Moisture ≥ 70%     | “watering”   |
| “no rain”                 | Moisture ≤ 70%, Light ≤ 2,000 lux, Temp ≤ 35 Celsius | “closing roof” |
| “rain” or “storm”         | Moisture ≥ 70%     | “wait for raining” |
| “rain” or “storm”         | Moisture ≥ 70%     | “opening roof” |

From Table 1, it can be observed that an optimum value of 70% has been ensued for moisture, 2000 lux for light and 35 degrees Celsius for temperature. As stated in the paper, these values are that of cabbage. Therefore, different plants have different optimum conditions. The decision tree model has been devised to allow the author to take decision for watering the corps. The author uses node.js library by the author is an integration of machine learning libraries. However, the node.js library cannot be considered as full-fledged processing system like Azure Machine Learning since it does not provide an array of algorithms which can be used to generate the prediction. Microsoft Azure Machine Learning provides different algorithms that are readily available and the appropriate one can be chosen and configured or adapted to suit the requirement. Furthermore, the author has not mentioned the type of soil which they are experimenting. The type of soil is important since different type of soils required different types of irrigation process. A notification alert could also have been implemented so as to notify the user about any particular changes in the crop area. Since the author is dealing with graphs, a web application could have been developed to have a better view of the sensor values. In [5], the paper proposes a smart irrigation system with the use of IoT together with smart technologies. Taking into consideration the wrong usage of water in the southern Algeria, a smart irrigation system that can be controlled and monitored has been devised to manage the usage of water more efficiently. The irrigation system is endured to be cost effective and detailed. Wireless Sensor Network (WSN) and IoT technologies have been to developed the system. Constrained Application Protocol (CoAP) and web application have been used to complete the implementation of the system where the web application was mainly for controlling the irrigation through the internet.

**Critics:** The project was mostly based on the network connections, it has focused more on establishing the WSN. Though it is good to explored different kinds of ICT technologies, the system could have been simpler. This is because the system can target the majority of the population with its simplicity. A database system could also have been established if historical sensor data need to be viewed later on. In addition, the system could have used microcontrollers or microprocessor to make
the connection between the irrigation area and the mobile application easier. In[6], the work is based upon an efficient and friendly Arduino-based automatic irrigation system which make use of Android smartphone for remote control. The prototype consists of a soil moisture sensor that gives a voltage signal proportional to the moisture content in the soil which is correlated with a fixed threshold value retrieved by inspecting various soils and explicit crops.

II. PROPOSED SYSTEM ARCHITECTURE

In this section, the proposed system architecture is elaborated. Figure1 shows an architectural overview of the proposed system. The proposed architecture for the Smart Irrigation and Monitoring System has been divided into 6 main components namely:

Physical Environment(A), AzureIoT Hub(B), StreamAnalytics(C), AzureSQLDatabase and PowerBI(D), Azure Machine Learning(E), and the mobile application(F).

Physical Environment

The first part is the plant section where sensors (DHT22, YL-100 Soil Moisture) are connected on a WeMos Board[10] for sensing the plant surroundings. The DHT22 is a cost-effective sensor that is used for monitoring both the air temperature and the humidity of air[11]. YL-100 soil moisture sensor[12] is used to measure the soil moisture content. The moisture sensor outputs a high level when the soil lacks water otherwise it outputs a low level. Besides, the sensitivity of the soil moisture sensor is flexible and unlike other soil moisture sensors, the YL-100 sensor does not require any additional convert or to operate. As illustrated in figure1, the sensors feed in raw data from the plants environment to the WeMosBoard which then sends the raw data from sensors to Azure IoT Hub.

Azure IoT Hub

The Azure IoT Hub is a service that allows bi-directional communication among devices. The Azure IoT hub acts as a middle man between different services and the physical environment, that is, device to cloud communication and vice versa. It receives updated data from sensors regularly. In general, the job of the Azure IoT Hub is mainly for monitoring every IoT devices and link them together.
Stream Analytics
Stream Analytics is a service offered by Azure and is a mandatory path to pass on data from the Azure IoT Hub to the Azure SQL Database. Intrinsically, the core feature of this service is to provide the flexibility of streaming millions of records per second. The raw data is sent in the form of JSON format misstructured into a tabular form so that it can be stored into the Azure SQL Database.

Azure SQL Database and Power BI
At this point, the data found in the database needs to be transformed into a more user-friendly representation as farmers will not be understand SQL queries. Hence to cater for this problem, PowerBI is used to reconstruct the data in to a visual representation such as a graph.

Azure Machine Learning
The Azure machine learning is the core logic of the proposed system. In general, a data set is needed to train the machine to find patterns in the data in order to decide whether to irrigate or not. For better precision, Open weather map.com API is included with the aim of knowing when the water pump needs to be opened. The pseudo code in figure2 gives a simple illustration on how the machine learning system works.

IF(Weather = "Rain")
THEN “No Irrigation”;
ELSEIF (Weather=”No Rain” AND Soil Moisture<”Threshold Value”)
THEN “Irrigate”;
ENDIF.

Smart Farming Data Module
The data set is input in this module. Basically, it contains a variety of data and need to be clarified to obtain the desired data, that is, the data for soil moisture, humidity and temperature. The clarification is performed by removing null dataset. The dataset chosen is based on a chart.

Select Column in Dataset Module
This module is implemented in the chain so as to be able to train the machine. Specific column has to be selected in order to do the prediction on a specific parameter.

Split Data Module
The split data module is divided into two parts: mainly the Train Mode land The Score Model (See section 3). The Train Model module contains 70% of the data set for the machine learning. The required pattern is trained in this area and to do so, an algorithm needs to be derived for it to know which data the pattern should consist of. The algorithm used is known as a "Two-class boosted decision tree" and lies under the "two-class classification" algorithm structure.

Figure 4 shows the different algorithms for the "two-class classification" algorithms. It can be
observed that there are nine types of algorithms in this category and they differ from each other. As illustrated in figure 4, the "Two-class SVM" has hundred features and provides a linear model, the "Two-class decision forest" provides accuracy and fast training and the "Two-class neutral network" provides accuracy in addition to long training time.

Fig 3: Training Experiment of Machine Learning

Score Model Module
After training the dataset, the machine learning checks whether the predicted value is accurate enough. To do so, as core model module needs to be set and it is based on the 70% dataset that has been provided from the Train Model. A random value is passed into the machine learning and is linked to remaining 30% of data to perform the prediction.

Evaluate Model Module
This module has been included during the testing mode and is intended to provide a more user-friendly representation of the results predicted.
Mobile Application
After processing all the raw data on the cloud platform, the user would be able to view the predictions through a mobile application. The interface of the mobile application is shown in figure. The mobile application contains a navigation drawer that has as default a weather forecast page and provides the farmer with a list of choices. The Arduino Registration allows the user to register his device for recognition purposes on the Azure IoT Hub and the farmer is provided with a unique Key. This registration is important in cases where multiple Arduino boards will be used and hence each of the semi-controllers need to have their own identification. The Sensor Readings and Graphs are used to view statistical data that has been processed. Lastly, in the Action menu the prediction from the machine learning is displayed and options to open the water pump with the switch ON/OFF button is provided. Xamarin Studio has been used to develop the mobile application as it is cross platform and hence it can target a maximum number of users. Broadly, the mobile application notifies the user to open the water pump if irrigation needs to be done. The user will have the flexibility not to irrigate if he feels that it is not necessary to irrigate even if he has been notified to open the water pump. Assuming that the user chooses to irrigate, once the button “ON” is clicked on the mobile application, the signal will be sent to Azure IoT Hub which will in turn forward the signal to the WeMosboard to open the water pump for irrigation.

The Web App
The web app is where the functions from the PowerBI will figure in. The web app has been designed using open source web design template. The reason for choosing open source is simply because it is free and we are exposed to the source code and hence we only need to mend the template as per our requirements. The web app has been developed in the water fall development process. Figure 6 shows a picture collage of the web app. The web app will consist of the PowerBI where the graphs of the sensor data appears. The graphs will be fully discussed in section 4. The "sensors" part in the web app contains information about the sensor. By hovering on the pictures, the latest sensor value will be displayed.
III. CONCLUSION

A Smart Irrigation and Monitoring System has been proposed so as to reduce wastage of water and to automate the irrigation structure of large areas of crops. The system mainly monitors the behavior of soil moisture, air humidity and air temperature and see how it contributes to evaluate the needs of water in a plant. The system uses machine learning and compares actual values obtained from sensors with a threshold value that has been fed to the machine learning for analysis. After this process, the machine learning cross checks the result obtained with weather forecast and then decides whether irrigation needs to be done or not. The farmer receives a notification on his smartphone and he can choose to turn on the water pump with a button click. Moreover, the system has a web app and is helpful if ever the farmer wants to see the statistical sensor data and assess the change in sensor readings throughout a time period. Furthermore the system can be calibrated for different type of plants, that is, the user is provided with al is to plants “choices in his web app and mobile app. With this the user can choose the specific type of plant that is being cultivated and obtain a more precise threshold value and thus a more accurate irrigation prediction. Besides an SMS system can be integrated if incase there is no internet connection. With this, the user would be notify about the prediction via an SMS and he can choose to switch on or off the water pump by replying to the SMS receive.
REFERENCES

I. Human Resource Development Council, 2012. A Study on Labour Shortage in the Agricultural Sector in Mauritius, October 2012.

II. Mauritius Meteorological Services 2017, Climate of Mauritius. [Online] http://metservice.intnet.mu/climate-services/climate-of-mauritius.php

III. A.A.N. and K.D., 2016. Experimental investigation of remote control via Android Smartphone based automated irrigation system using moisture sensor. 3rd International Conference on Electrical Energy Systems (ICEES), 2016.

IV. T. Baranwal, N. and P. K. Pateriya, 2016. Development of IoT based Smart Security and Monitoring Devices for Agriculture.

V. International Conference - Cloud System and Big Data Engineering (Confluence), 2016.

VI. G.M.K., J.J. and A.M.G.S., 2015. Providing Smart Agriculture Solutions to Farmers for better yielding using IoT. IEEE Technological Innovation in ICT for Agriculture and Rural Development (TIAR), 2015.

VII. Elementz online 2016. Programming ESP8266 WeMos – D1R2 using ArduinoIDE, Random Codes-ElementzTech, 2016

VIII. Microsoft, 2017. Machine learning algorithm cheat sheet. Microsoft Azure. [Online] https://docs.microsoft.com/en-us/azure/machine-learning/machine-learning-algorithm-cheat-sheet

IX. Dynamic, 2017. Soil Moisture Range Chart, TurfIrrigation.

X. GitHub 2017 Azure/azure-iot-sdk-csharp. [Online] https://github.com/Azure/azure-iot-sdk-csharp/tree/master/tools/DeviceExplorer

XI. V. Proag, 2006 Water Resources Management in Mauritius, European Water, E.W. Publications, 2006.