Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 6.017

IJCSMC, Vol. 8, Issue. 2, February 2019, pg.62 – 66

Wireless Sensor Network (WSN) Predictable Weather Station

Dr. Sefer Kurnaz
Electrical and Computer Engineering, Altin Baş University, Istanbul, Turkiye
sefer.kurnaz@altinbas.edu.tr

Laith Mohamned Salim
Electrical and Computer Engineering, Altin Baş University, Istanbul, Turkiye
laith79@gmail.com

Abstract— Weather prediction one of the projects that get a lot of attention in these days. There are many challenges generated during implementation of the system especially when needed to collect data from a number of sensors and that means there are sending data from the outdoor sub-station to the indoor sub-station. In this work we built a weather station that able to predict that the atmosphere is playable or not for helping people to make a decision. This means that we will receive data from the sensors and then examine the data by using one of the machine learning algorithms. The weather station will measure the temperature, humidity, rainfall and detects wind.

Keywords: Weather, Arduino, Sensor, and Decision Tree.

I. INTRODUCTION

Weather conditions affect our daily activities very effectively. Sports is one of the most common events associated with the weather. Where the most popular types of sports be outside. The weather conditions can even affect the health if the sport exercised in bad conditions. In addition to the sport, there are a lot of activities which depending on the weather such as a picnic, travelling, and shopping.

There are several methods for collecting the data from the weather. One of these solutions is collecting the data and send them by wires to a central station. This way is very costly since it requires places for the weather terminals. The second way is used the wireless sensor networks. These networks are developed in a number of different areas for the aim of collect data and then send it by wireless. This solution offers a high resolution on the other hand it has many disadvantages such as low accuracy, the lifetime is limit and the robustness also limit which is the disadvantage of the most wireless sensor networks. The last way offers using the weather station with low dimensions. Some of them are static while the other is mobile. Usually placing the sensors in a board with simple size such few inches. The CPU is responsible for processing and opening interface for communication. There are laptops or desktop computers to control the small weather stations locally [1].

The rest of the paper is organized as follows: Section II presents the related works, Section III describe the proposed weather station (Hardware and Software), Section IV discussion of the results and data obtained from the system developed in this study, and finally, Section V provides concluding and discussion of possible future work.

II. RELATED WORKS

There are many types of research have been done to build a weather station and weather forecasting. The paragraphs below explain some of these works.

In work [1] propose weather station using sensors for measuring the pressure luminosity. Also the measure the temperature and the humidity and placing these sensors in a board with a microcontroller. The station can be controlled by mobile phones through the SMS service.

Reference [2] present an automated weather station for real-time and local measurements, based on an embedded system that continuously measures several weather factors such as temperature, humidity, barometric pressure, wind speed, wind direction, and rainfall. This weather station consists of two parts which are located indoor and outdoor and connected together wirelessly.

Reference [3] aims to build a weather station as a second prototype for measurements the conditions of the weather such as the temperature, pressure and the humidity. they used variance analysis to validate the weather station operation. also, the used an experimental design which called r&R. a number of sensors selected such as TMP36, RHT03, and BMP085 with Arduino UNO board in the weather station and they used a digital hygrometer.

Reference [4] develop weather station for monitoring weather in a real-time. they used a mobile application depending on the Automatic Weather Station (AWS). The system contains several sensors connected to the AWS to collect data. they stored the data on the web server. The mobile application which using Android system able to read the data and displays it from the web server.

Reference [5] shows a low-cost weather station capable of measuring the temperature of the air. Also it measures the humidity, pressure, and able to detect the wind speed and in which direction and the amount of the rainfall. A LiPo battery and four solar panels were used to power up the complete system. The weather station communicates through Wi-Fi connection allowing the user to access the data remotely.

III. THE PROPOSED WEATHER STATION

A. General Overview:

In this work, we will build a weather station that can read data from the weather by using sensors and make a decision to play a game or not based on weather conditions. We will use a decision tree. Decision tree algorithm one of the most used methods in machine learning. It is suitable for both regression and classification problems. The weather station will contain two sub-stations. First one with the sensors will place outdoor to get real weather data. The second sub-station will place indoor to receiving the data from the outdoor station and make the decision. Figure 1 presents an overview of the weather station which we designed.

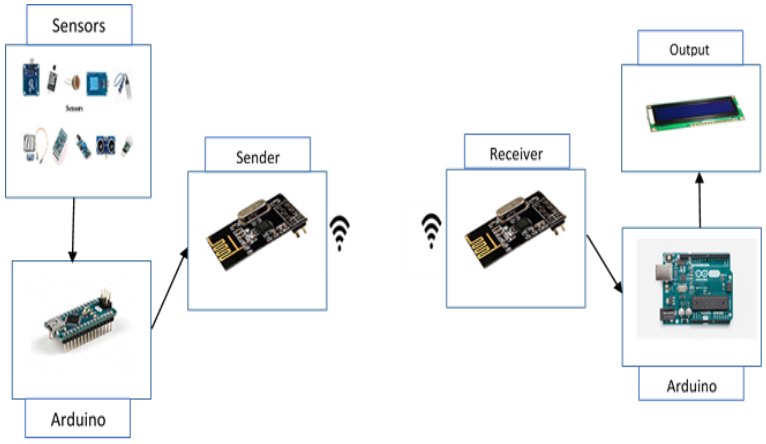

Fig. 1. Weather Station Structure.

B. Hardware Part:

As we mentioned previously our weather station contains two parts. In the next few paragraphs we will explain them in details.

a) Sensors: there are three sensors in this part of the weather station: dht11 temperature and humidity sensor, rain sensor, and LDR light sensor. They are responses for gathering information about temperature, humidity, rain, and light (sun). also, there is a dc motor used as a wind sensor. The sensors are selected in order to match the data in the training set which will be explained later.

b) Arduino Nano: this Arduino will be responsible for running the outdoor part of the weather station and also will collect the data from the sensors and convert them to values that can be understood by the human then sending it to the indoor station using the transceiver.

c) *NRF24L01:* it is a receiver and transmitter wireless module. It is responsible for sending the data from the outdoor part to the indoor part of the weather station. On the other hand, it is responsible for receiving the data in the part which will be indoor. It is working with 2.4 GHz Bandwidth with Band Rate between 250 kbps and 2 Mbps. The range of this transceiver up to 100 m. the max current draw is 12 mA with an operating voltage range between 1.9 and 3.6 V.

d) *Arduino UNO:* this Arduino will be responsible for running the indoor part of the weather station and also will receive the data from the outdoor part. This Arduino will run the machine learning algorithm to make the decision about the weather. The results are displaying on the LCD screen.

e) *Power Source:* for each part, outdoor part and indoor part of the weather station, we will use a lipo battery with 7.4V and 350 mAh to supply it with power.

Fig. 2 showing the indoor and outdoor weather station parts:

Fig. 2. Outdoor and Indoor Parts of our Weather Station.

C. Software Part:

In the software part we will talk about the decision tree algorithm. For building our decision tree, we used the playing game dataset. It is famous and very used in most machine learning solutions. It gives a yes or no decision depending on the conditions of the weather. The table below shows the playing game dataset [6]:

TABLE 1 DATASET

Windy	*Temp Numeric*	*Humidity Numeric*	*Outlook*	*Play*
FALSE	83	86	overcast	yes
TRUE	64	65	overcast	yes
TRUE	72	90	overcast	yes
FALSE	81	75	overcast	yes
FALSE	70	96	rainy	yes
FALSE	68	80	rainy	yes
TRUE	65	70	rainy	no
FALSE	75	80	rainy	yes
TRUE	71	91	rainy	no
FALSE	85	85	sunny	no
TRUE	80	90	sunny	no
FALSE	72	95	sunny	no
FALSE	69	70	sunny	yes
TRUE	75	70	sunny	yes

To build the decision tree, first we calculated the entropy and information gain for each feature in the dataset. The dataset contains 4 features (outlook, temp, humidity, and windy) to figure out which attribute that best classify the training data Reference. The equation of entropy and information gain explained below:

$$H(S) = \sum_{c \in C} - p(\mathrm{c}) \log_2 p(c) \qquad (1)$$

Where, S is the current dataset for which entropy is being calculated. C {yes, no} is the set of classes in S. P(c) is the proportion of the number of elements in class c to the number of elements in set S.

The equation of information gain depending on the entropy will be in this form:

$$IG(A.S) = H(S) - \sum_{c \in C} - p(t)\, \mathrm{H}\, p(t) \quad (2)$$

Where, the term H(S) is referred to the entropy of the dataset S. T refer to the subsets which splitting from the S. p(t) is the ratio of the elements in t to the total elements in the dataset. Finally, the entropy of the subset t denoted by H(t).

Depending on the entropy of the dataset which equals 0.940, the results of each attribute are shown below:

TABLE 2 INFORMATION GAIN OF THE DATASET

Outlook	Temperature
Info: 0.693 Gain: 0.940 – 0.693 = 0.247	Info: 0.911 Gain: 0.940 – 0.911 = 0.029
Humidity	**Windy**
Info: 0.788 Gain: 0.940 – 0.788 = 0.152	Info: 0.892 Gain: 0.940 – 0.892 = 0.048

So our root node is Outlook. After repeating the same steps for the sub-trees, we get the following tree:

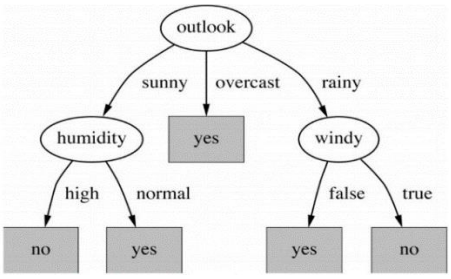

Fig. 3. Final Decision Tree [6].

After building the decision tree, we built our program using the Arduino development environment. Our code receiving data from the sensors. Then comparing the incoming data with the decision tree to give us yes or no decision.

We convert the incoming data to be suitable with the dataset. Also, we convert the temperature in the dataset from the Fahrenheit to Celsius because it is easier to read it by the human in Celsius rather than Fahrenheit.

The incoming data from the Humidity, Rain, and Sunrise sensors divided into ranges as we said before to be more suitable. Below is the range for each variable:

- Humidity: If sensor reading > = 85 → High.
- Sunrise: If sensor reading > 300 → Sunny. Else If sensor reading <= 300 → Overcast.
- Both the wind and rain sensors are digital which mean their outs will be 0 or 1 and the 1 means positive while the 0 mean negative.

We upload this program to the Arduino in the indoor sub-station.

IV. RESULTS

In this section we will show the data which received by the indoor station. Also, we will show the decision which made depending on this data. We took 10 readings from the weather station. some of the readings have different weather conditions. While the other similar in the conditions. Table 3 is showing the data and the decision:

TABLE 3 RESULTS

Rain	*Light Sensor*	*Temp Numeric*	*Humidity Numeric*	*Wind Sensor*	*Play*
0	195	17	71	0	Yes
0	199	17	55	1	Yes
1	143	19	69	0	Yes
1	159	17	74	0	Yes
1	160	18	77	0	Yes
1	128	19	65	1	No
0	528	18	54	1	Yes
0	627	17	82	0	Yes
0	508	18	93	0	No
0	580	18	93	0	No

V. CONCLUSION AND FUTUR WORKS

This work describes a weather station for making playing decision using Arduino and some sensors based on wireless NRF24L01 communication model. The advantage that the incoming information is easy to understand because we are showing all the received information and the decision on the LCD screen. We used a decision tree algorithm with playing game dataset for decision making. As a result, our weather station can work directly and measuring the temperature, humidity, sunlight and able to detect rain and wind from any environment.

For the future work we advise to use long-range communication method to increase the range of distance for sending data. Also we advise using the weather station to make predicates about the weather for incoming days not only the current time.

REFERENCES

[1] Mircea Popa, "Embedded Weather Station with Remote Wireless Control", 19th Telecommunications forum TELFOR 2011.

[2] Mohamad Farhat, Mohammed Abdul-Niby, Mohammed Abdullah, and Abdelbaset Nazzal, "A Low Cost Automated Weather Station for Real Time Local Measurements ", Engineering Technology & Applied Science Research, 2017.

[3] Gabriel Piñeres-Espitia, Alejandro Cama-Pinto, Daniel De La Rosa Morrón, Francisco Estevez, And Dora Cama-Pinto, "Design of a low cost weather station for detecting environmental changes ", Revista Espacios, 2017.

[4] Aris Munandar, Hanif Fakhrurroja, Muhammad Ilham Rizqyawan, Rian Putra Pratama, Jony Winaryo Wibowo, and Irfan Asfy Fakhry Anto, "Design of Real-time Weather Monitoring System Based on Mobile Application using Automatic Weather Station ", International Conference on Automation, Cognitive Science, Optics, Micro Electro Mechanical System, and Information Technology (ICACOMIT), 2017.

[5] Diego A. Aponte-Roa, Luis Benitez Montalvan, Christopher Velazquez, Albert A. Espinoza, Luis Feliciano Velazquez, and Ricardo Serrano, "Evaluation of a Low-Cost, Solar-Powered Weather Station for Small-Scale Wind Farm Site Selection ", IEEE, 2018.

[6] https://medium.com/deep-math-machine-learning-ai/chapter-4-decision-trees-algorithms-b93975f7a1f1.

