International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 02 | Feb 2020 www.irjet.net p-ISSN: 2395-0072

Machine Learning for Weather Prediction and Forecasting for Local Weather Station using IoT.

Manan Praful Raval[1], Shamli Rajan Bharmal[1], Fatima Aziz Ali Hitawla[1]
Prof Pragya Gupta[2]

[1]Department of Electronics Engineering, K. J. Somaiya College of Engineering (Autonomous), (Affiliated to University of Mumbai), Mumbai, India)
[2]Assistant Professor, Department of Electronics Engineering, K. J. Somaiya College of Engineering (Autonomous) (Affiliated to University of Mumbai), Mumbai, India)

---***---

Abstract - *The aim of our project is to monitor several aspects of weather using IoT based smart system and to predict the future values. The main advantage of using this concept is it helps in monitoring weather of a local area and thus it helps in developing a microclimate system. Based on the monitored database it predicts the future weather of a particular zone thereby making the result more accurate and relevant for the zone. This system a portable alternative that can be adapted in broader applications. It is better than the existing websites which displays collective weather information whereas our system is more specific to the changes in weather adhering to a particular area. It can find its application in corporate offices, hospitals, educational premises such as schools, colleges and university campuses to switch to optimum temperature as per requirements. Microcontroller is interfaced with Wi-Fi module which helps to send the sensed data to the open source platform. IoT provides a platform to display and store the parameters in cloud which is extracted in the form of CSV file. The extracted data is fed to a Machine learning model that uses Time series analysis algorithm called as ARIMA. This model predicts future values of the various weather parameters which are then displayed onto the server.*

Key Words: ARIMA, IoT, Machine Learning, Time Series Analysis.

1. INTRODUCTION

IoT based Weather monitoring and forecast system can be used in a variety of places including work places, schools, colleges, offices for monitoring temperature, humidity and pressure and displaying the results on a user-friendly website. This makes IoT based weather monitoring and forecast project extensively useful in various organizations. The sensors monitor various parameters in the environment for example-Temperature, Pressure and Humidity. The sensed data is then sent to the open source platform through wi-fi module. The data is displayed on the Thing speak channel in the form of graphs. Also, the data stored in the cloud is extracted in the form of CSV file. Then the data is fed to machine learning model which uses Time series analysis algorithm, it processes the data into data frames. The model helps to display the forecasted values on the server. This is an effective way of monitoring weather of a specific zone in order to take preventive measures in case of emergencies or any hazards. The development of IoT-based commands using this project leads to the development of another aspect of technology that can deal with control of appliances and gadgets using internet[1][2]. It is a mere beginning of new age technology which aims at making lives simpler. The scope of IoT- based weather station is wide in regions where ease of access has priority. The IoT based Weather Informative System will be proposed to Real time Applications. It doesn't need any data centres physically because of we are creating a data Server in cloud so that it doesn't require any physical data centre further. So, it reduces the cost of equipment. Many of the innovative researchers are interested towards the IoT based Real time applications. So, this system will help the researcher for their further investigation of weather details. The IoT based Weather Informative System not only displays the weather parameters like temperature, altitude, humidity and pressure etc., but it also displays the weather location, Industry, Time and other weather information from this we can forecasts the weather details. Scope of this project is not just limited to home automation, but it could also to smart city applications as well as industrial procedures[2]. In an emerging world of technology and science, IoT is the newest paradigm. It has huge scope in future. Making an optimum use of IoT and being a developer instead of a user, utilizing it for our routine activities indeed brought innovation out of us[1].

2. BLOCK DIAGRAM

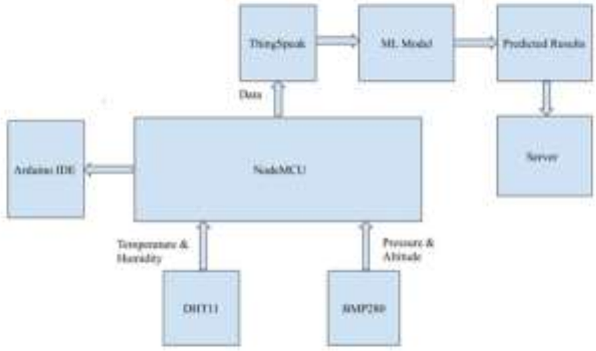

Fig -1: Flow chart (IoT)

The above flow chart in Fig 1 explains the proposed technique. The DHT11 and BMP 280 sensors sense the following weather parameters Temperature & humidity and Pressure & Altitude respectively. The sensed values are then coded into a microcontroller named NodeMCU which act as a intermediate interfacing element that communicates between the sensors and Internet. The sensed values are pushed to an open source cloud storage Thingspeak.com. The data is extracted in the form of CSV from the cloud storage. This data is then fed to a designed machine learning model named ARIMA (Auto Regression Integrated Moving Average) which is a Time Series Analysis algorithm used to predict and visualize patterns and trends of the data.

The predicted data from the machine learning model is then displayed onto a web application designed using Python as a backend, Django as MVC and HTML, CSS, jQuery and Bootstrap as front end. The website had been opened on a local server.

3. HARDWARE DESCRIPTION

3.1 NodeMCU (Microcontroller):

NodeMCU is an open source development board. It allowed us to program the ESP8266 Wi-Fi module with the simple and powerful LUA programming language or Arduino IDE. With the help of a simple code we established a Wi-Fi connection and we could define input/output pins according to the user defined needs exactly like Arduino, turning ESP8266 into a web server and many more. It is the Wi-Fi equivalent of ethernet module. Thus, providing us Internet of things (IoT) real tool. The ESP8266 is a low-cost wi-fi chip developed with TCP/IP protocol. NodeMCU development kit provides access to the GPIOs of ESP8266.. It combines features of Wi-Fi access point and station along with microcontroller. These features made the NodeMCU extremely powerful tool for Wi-fi networking which was therefore used by us as an access point and/or as a station, host a webserver or connect to internet to fetch and upload data

3.2 Sensors:

In this project we have used a DHT11 sensor in order to sense temperature and humidity of the surrounding. DHT11 is a electronic brick sensor which calibrates to give digital signal output.It has a single bus operation, its extremely small in size and its low consumption enables it to be used in vast application like HVAC, automotive, weather stations, dehumidifier, etc. The BMP280 is an absolute barometric pressure sensor especially designed for mobile applications.

4. SOFTWARE DESCRIPTION:

4.1 Open Source IoT Analytics and Cloud Storage:

Thingspeak is an open source IoT Analytics and Cloud Storage platform. It is an API to store and redeem data from things using a http protocol over the internet or via a local server. ThingSpeak helps to create a sensor logging application, location tracking application and a social network with status update.

We have configured our Thingspeak account with the Microcontroller in order to publish the data sensed by the hardware to the Analytics dashboard of Thingspeak.

We can extract the data from the Thingspeak dashboard in the form of data in CSV for further use of the sensed data.

4.2 ARIMA (Machine Learning Model) :

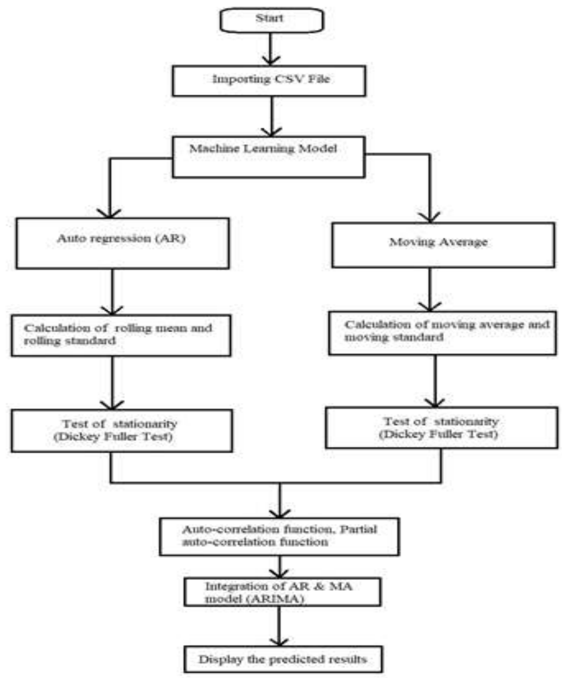

Fig -2: Flow chart (ARIMA Model)

The flow char of the ARIMA model in Fig 2 is explained below in detail:

Auto Regressive model:

For time series analysis, auto regressive model is the most important part. In the above model, we get auto regression stats in the function: rolling mean and rolling standard. Further we carry out Dickey fuller Test for testing the stationarity of the function: The closeness between p values approaching towards zero leads to efficient results.

Autoregression is a model which uses a relationship between observation and some good number of lagged observations.AR model is used to calculate the value of p for the order of ARIMA. p is the number of lag observations included in the model, it is also called a lag order[4][6].

Integrated:

In the integrated model, we use the difference of raw observations i.e we calculate the degree of differencing d on the basis of subtracting an observation with an observation from the previous step. When a trend is not stable or organized ,it boils down to the fact that the trend is not stationary[5]. The integrated model helps us to apply degree of differencing in order to make the trends stationary for forecasting.

Moving Average:

In the moving average model, we convert the dataset into logarithmic function. The log scale function is used to make the high skewed functions into less skewed functions. For the moving average function, we calculate the values moving average, moving standard. And then similarly carry out the dickey fuller test for testing the stationarity. MA model is used to calculate the value of q for the order of ARIMA. q is the window of moving average[7].

Dickey fuller test:

Dickey Fuller test is carried out in order to calculate the most important factors viz p-value and critical value. One of the main purpose of dickey fuller test is to test the stationarity and hence the smaller the p-value, the more likely it's stationary. In the first dickey fuller test we got a p-value around 0.001789 as shown in Fig 3

Fig -3: Dickey fuller test results(Non-stationary)

To get stationary data we used logarithm in order to decrease the p-value further which is shown in Fig 4 [6][7].

Fig -4: Dickey fuller test results(Stationarity Test)

Autocorrelation and Partial Autocorrelation:

ACF and PACF functions are used to find the p, d and q values of ARIMA model to be calculated for the input values shown in Fig 5.

Arima model is a very useful tool in order to calculate the predicted value of the input values. The order of ARIMA model is (p,d,q) where p is achieved from acf function and q is achieved from pacf function. ARIMA Model is a integration of Auto-regression and Moving average of the function

Fig -5: Auto Correlation and Partial Auto Correlation

4.3 Software Integration:

In order to cast the results obtained from the machine learning model, we have integrated a web framework called Django. Django is a web framework which adheres to the MVC (models, views, control) architecture. We have used Django as a mediator to integrate the model built in Python with multiple web technologies. The multiple web technologies that we used are HTML, CSS, Bootstrap and jQuery. We imported the functions of the machine learning model and its result into the models of Django. The models of Django were then fed to the views in order to integrate the functions of the model onto the web page. We built the front end of web page using HTML, CSS and Bootstrap. For backend integration we used JavaScript, Django and Python.

4.4 Database:

We created a database which has four columns for date and time, temperature, humidity and air pressure. The data is being fed from the Thingspeak cloud into the database. The machine learning model extracts data from the database in the form of data frames by using an inbuilt python library named pandas. The extracted data is coming in the form of comma separated values (csv). We are maintaining a dataset of hourly interval of everyday for a span of one year. We used the database rigorously, as machine learning increases the productivity and efficiency of the model when it has large amount of history data to predict future values. More the data, greater is the optimization.

4.5 Website & Server:

An user interface has been created for the users in order to monitor the live data as well as to display the future data for the coming two weeks as well. The front end website has been designed using front end technologies: HTML, CSS, Bootstrap and jQuery. The user has the privileges to select the date of which they want to access the data for. The back-end technology scripted in JavaScript, Python and Django, retrieves the data from the model for future data.

The website has been dynamically coded in order to make the website completely user friendly. The website has been hosted on a local server.

5. RESULTS:

5. 1 ARIMA Model:

The results provided by machine learning model are displayed in fig 6. The window of number of days for prediction can be refactored in code.

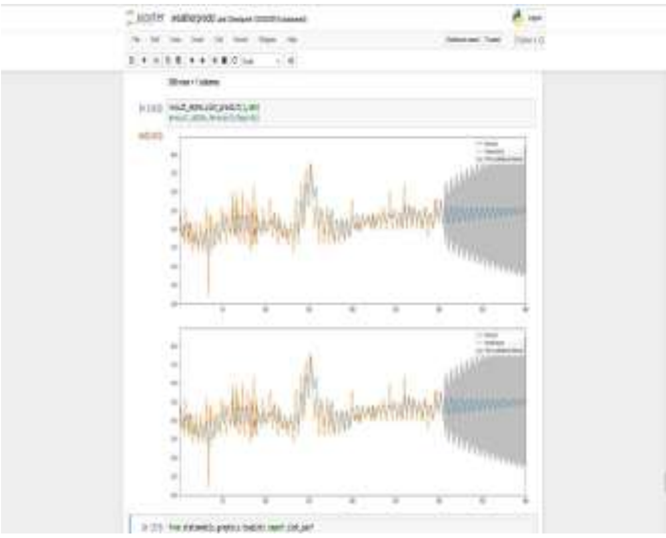

Fig -6: Model Results

We achieved a model accuracy of 96.5 % when carrying out test trainings on big dataset.

5.2Cloud:

The microcontroller NodeMCU would process the sensed data and send it over to the cloud platform named Thingspeak.com.

Fig -7: Cloud dashboard

The results displayed in fig 7 are user friendly dashboards designed in order to get a great insight of the valuable parameters. We used Thingspeak API in order to send the data over to the cloud. The four parameters that we have chose to display are Temperature, Pressure, Humidity and Altitude.

5. 3 Interfacing:

The microcontroller has been been integrated with the sensors in order to send the data from sensors on to the cloud. The interfacing of hardware with cloud has been done using Arduino IDE. We coded the program in C to interface the hardware with the cloud. The inbuilt wifi module ESP8266 in NodeMCU helps to communicate with the desired server (which is thingspeak.com in this case) to send the data from microcontroller to the server. The results have been displayed below in Fig- 8.

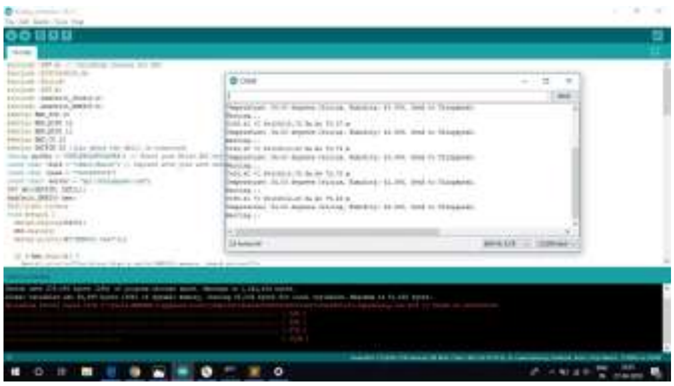

Fig -8: NodeMCU Result

5.4 Website:

The website consists of following tabs.

a) The first tab in Fig-9 displays the live monitoring of the weather parameters in a single dashboard. It updates continuously at an interval of 60 seconds.

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 02 | Feb 2020 www.irjet.net p-ISSN: 2395-0072

Fig -9: Web Application

b) The second tab in Fig-10 displays the future weather where in the user has the liberty to select a date within the next two weeks from the day the user is using it. On filling the date, the results are displayed onto the web page. This page is generically coded to fetch the values based on the user input and displaying it from the models.

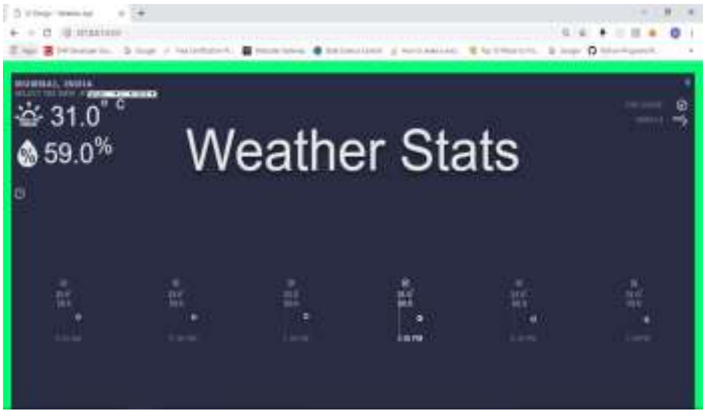

Fig -10: Future Predicted results on User Interface

A coherent approach to achieving prediction and forecasting of weather has been done using Time Series Analysis as a machine learning algorithm. Also IoT based Local weather monitoring system has been developed which can be a boon for the coming world of smart city. The results and accuracy of the prediction model has been robust. A prototype has been designed in order to elaborate and explain the proposed model and the results are recorded.

REFERENCES

[1] "The Internet of Things" by Samuel Greengard Publication, 2015.

[2] "Getting started with Internet of Things" by Cuno Pfister Publication, 2011.

[3] "Python Machine Learning: Machine Learning and Deep Learning with Python" scikitlearn and TensorFlow by Sabastian Raschka 2nd Edition Publication, 2011

[4] https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/

[5] https://www.timeanddate.com/weather/india/mumbai/historic?month=3&year=2018.

[6] https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/.

[7] https://www.datascience.com/blog/introduction-to-forecasting-with-arima-in-r-learn-data-science-tutorials.