Employing Machine Learning and Internet of Things for Malaria Outbreak Prediction in Rwanda

Janvier Niyitegeka[1], Didacienne Mukanyiligira[2], Said Rutabayiro Ngoga[2], Emmanuel Masabo[2], Louis Sibomana[2] ,Raymond Ndacyayisaba[3]

[1] , [2] University of Rwanda, College of Sciences and Technology.
[1] , [3] Rwanda Polytechnic, IPRC TUMBA.
[1] nijas2012@yahoo.com

Abstract. Malaria is a threatening disease which is caused by a bite of female mosquitoes called anopheles and when it is not discovered at its earlier stage, it can put the life of many people at risk and even reduce the workforce of the country. However, its rate of transmission can be decreased if the information regarding the development of these mosquitoes are made available in due time. However there is a lack of real time information about Malaria spreading to help the Ministry of Health to know the development of malaria mosquitoes relatively to environmental conditions and take the required measures for fighting against the spread of this disease by providing early warning to the hospitals and health institutions to purchase the medicine on time and reminding the citizens to use mosquito nets accordingly.

The current study mainly aims to apply machine learning and Internet of Things technologies to help the Ministry of Health(MoH) to have access on the development of malaria mosquitoes and provide early warning information across citizens, hospitals, health institutions and individuals to be prepared accordingly. For modelling the dependency of malaria transmission, we have tested different machine learning classification algorithms for optimizing the prediction accuracy. The data used include the environmental climate and malaria data recorded by METEO Rwanda and Ministry of Health respectively in the period of 8 years (2012-2019) from Bugesera and Huye districts the most malaria endemic district in Rwanda.

The results show that the Artificial Neural Network algorithm could perform better than other algorithms tested with 93.9% and 88.2% of training and testing accuracy respectively in Bugesera district ,and 88.9% and 62.5% for training and testing accuracies respectively in Huye district. Secondly, an IoT based system was implemented to interact with the predictive model and view the results of prediction in the future on field sensors data via Smartphone, tablet or PC.

Keywords: Malaria, Mosquitoes, Machine Learning, Internet of Things, MQTT,Node-Red.

1 Introduction

Malaria is a serious disease, sometimes fatal, spread by mosquitoes namely called by plasmodium and being developed when the climate conditions are favourable for them to breed [1, 2]. The Rwanda Annual Health Statistics Booklet of 2016 showed that the number of malaria cases confirmed from 2012 up to 2016 equals to 481868 ,934484, 1597143, 2456091, 4637483 respectively and associated deaths were 459, 409,496, 489, 715 respectively where high percentage counted on the districts known to have a high malaria burden in Rwanda, mostly located in the eastern and southern provinces [3].

In [1], the report says that it is a must to reflect on the achievements made so far, the challenges encountered and take the measures to end Malaria by 2030 in African Region and Rwanda in particularly. It is this line the USAID report of 2018 showed that there is a good trend in fighting against this disease for reducing the deaths associated with it [4]. Apart from this initiative made so far, there is still a need to apply the ICT technology to generate and disseminate information that can be used by all stakeholders (Ministry of health, individuals) to take the informed policy decisions based on the climate changes.

In some African countries, different machine learning predictive models were undertaken by previous researchers for data analysis and for building malaria predictive models for helping the community to derive new information, making new better decisions and facing the events that are likely to happen in the future [2, 5–9]. Through these research studies, the researchers used different machine learning algorithms to predict the abundance of malaria mosquitoes using health and climate variables data on the geographical location like India, Mozambique, Ethiopia etc.

However, the result presented by these previous researches might be judged in different way when they are applied here in our context(in Rwanda) because the geographic locations, the vegetation index , habitation , social, economic, public health and political situation are different and these might impact differently the predictive accuracy [2]. Secondly, through this literature review, these researchers did not address the way forward to use the generated predictive models to make predictions in the future on new climate data collected by sensors from the field for easy human work. Therefore, we found to be very important to design and implement an IoT based real time data collection system that might benefit from the predictive model generated by the researcher to predict the abundance of malaria mosquitoes in the future based on the climate conditions. Therefore, the main objective of this paper aims at modeling the dependency of malaria transmission in Rwanda specifically in Eastern and Southern provinces the most malaria endemic provinces.

In this research, we have implemented and tested the efficiency of different Machine Learning (ML) algorithms such as K-Nearest Neighbors (KNN),Logistic regression, Random Forest, Gradient Boosting, Decision Trees and Artificial Neural Network for predicting the outbreak of the malaria transmission based on the climate variables change.These machine learning algorithms have been selected because of easily availability of their documentation and popularity in solving machine learning regression and classification problems. After confirming the best candidate of machine learning model to make malaria outbreak prediction, the IoT based architecture was used to make the prediction on real-time climate data from sensors.

2 Problem Statement

Malaria is a leading cause of many deaths in African countries and putting the entire population at the risk of being infected, where the high percentages are given to the children under five years and pregnant women. The malaria report of 2019 showed that 228 million of malaria cases were identified worldwide and the African countries accounted for 93% of total cases identified and 94% of total deaths [10].

Rwanda Government has shown a high commitment to put an end to the risk associated by the malaria by the end of 2030 as it is particularly shown in its Sustainable Development Goals (SDGs) [1]. The Government of Rwanda is collaborating with different Health organization for combating against this disease by reducing the numbers of death caused by malaria [4].
Through the ICT sector, different research on malaria cases have been undertaken for generating the predictive models to uncover the development and the spread of malaria mosquitoes in some African countries [8, 11]. Although these previous researches have been undertaken, the environmental data used has been collected by the independent institutions and the researchers have not addressed the methodology approach in which the generated predictive models would be operated with IoT systems for generating the predictions in real-time with minimal user intervention. Secondly, the predictive accuracies generated by the previous predictive models were different and sometimes too small and tend to over-fit. These differences in the prediction accuracies might depend highly on the quality of the collected data, methodology approach used by the researcher and how well the data were prepared.

Thus, we find to be very important to investigate further research that includes also an IoT based real time data collection and continue reviewing different machine learning algorithms because the prediction accuracy will depend on the quality of data collected, methodology approach used for analyzing the data, political situation and the social economic activities of any country [2].There is a need also to broaden research in different countries because the factors that contribute in development of the mosquitoes that cause malaria might vary from one country with another based on the geographic location and even habitation situation. So, we need to build our specific predictive model based on the real

situation of our country.
Moreover, currently there is a good development in Rwanda Ministry of Health where it using the drones for disseminating medicines for killing these mosquitoes in some valley at its earlier stage but there is a lack of the real time information about the spread malaria mosquitoes because these insects are seasonally and cannot be predicted by using the existing ecosystem. So, the outcomes of this research undertaken in collaboration of Rwanda Ministry of Health and METEO Rwanda could be used to put an end to the future risk that might be revealed.

3 Ojective of the Study

3.1 Main Objective

The main objective of this paper is to employ the Machine Learning and Internet of Things for Malaria outbreak prediction in Rwanda specifically in Eastern and Southern provinces the most malaria endemic province by using the climate variables such as Temperature, Relative Humidity, Rain Fall volume recorded by Rwanda Meteorological Agency and malaria data recorded by the Rwanda Ministry of Health and the population growth of targeting region.

3.2 Specific Objectives

1. Analyze the relationship between malaria mosquitoes abundance and the environmental factors such as temperature, relative humidity and rain fall that impact the development of these mosquitoes in Rwanda.
2. Build machine learning model to map the relationship between the malaria transmission rate and the climate data.
3. Evaluating ML predictive model for optimizing the malaria transmission outbreak prediction accuracy.
4. Implement an IoT based real time climate data collection system that uses the predictive model to make prediction.

4 Literature Review

Through [6], Thakur S. et al; have used an Artificial Neural Network (ANN) machine learning algorithm to predict the abundance of malaria mosquitoes using clinical and environmental variables and the geographical location of Khammam district, Telanagana in India. Here the predictive model generated by the training process was based on the data that have been gathered from the health centers of Khammam district and the climate data such as humidity, rain fall, temperature and vegetation recorded for the period of 1995–2014.The research showed that the rain fall volume increase the numbers of mosquitoes cases which might lead to malaria outbreak transmission and revealed that when the humidity is at

60%RH and temperature is 28 Degree Celsius these make the mosquitoes environmental conditions favorable to breed. However, the mosquitoes are impacted negatively when the temperature is above 30 degree Celsius and less than 16 degree Celsius. The results show that the models accuracy varies from location to location. However, the researchers did not provide the more information about the models performance on both training and testing data

In [2], Richard et al; used the monthly malaria data collected at provincial level by the Ministry of Thailand and environmental and meteorological factors between 1994 and 1999. The data collected include precipitation, temperature, relative humidity, and vegetation index to model the dependency between the abundances of malaria mosquitoes and the climate and environmental variables using neural network based machine based algorithm provided by Artificial Intelligence (AI). The results from this research study provided the training accuracy of of 72.8% and testing accuracy of 62.9% in the three provinces. However, this evaluation metric does not provide the full information about the model performance because the prediction accuracy alone cannot illustrate how the model performs on each class category.

Belay EnyewChekol et al in [8], tested different Machine Learning Algorithms such as Support Vector Regression (SVR) and Adaptive Neuro Fuzzy inference System (ANFIS) to predict the malaria cases in Ethiopia by using the environmental data and climate data such as rain fall, relative humidity, elevation, temperature and malaria cases recorded in 2013-2017. During of model evaluation, the R square score and Root Mean Square Error (RMSE) machine learning evaluation techniques were used for checking how well the algorithm is learning. The results show that SVR model achieved the RMSE values for training and testing of 0.04 and 0.12 respectively and 0.049 and 0.133 for ANFIS RMSE respectively.

The researchers in [9], evaluated various machine learning classification models for testing which best model suited for malaria cases prediction using meteorological data malaria cases data recorded during of six years. The dataset used in this study was divided into two independent sub-dataset with 80% for training and 20 % for testing. The overall results of machine learning algorithms showed that Gradient Boosting, Artificial Neural Network, Random Forest, Support Vector Machine perform best with predicting accuracy of 96.3% ,93.9%, 94.3 and 92.7% respectively by achieving the highest score of accuracy in predicting the malaria cases. Unfortunately, the researcher does not provide the complete information on the performance of the model on both training and testing data.

Orlando P.et al in [11], used the Support Vector Machine (SVM) and Random forest (RF) machine learning algorithms for predicting malaria cases in Mozambique. The dataset used during this study, contained the records of malaria cases and the climate data such temperature, rain fall, humidity collected during of ten years from 1999 up to 2008 in eight district of Maputo province on monthly bases. At the end the results of the study showed that the support vector machine algorithm perform better than random forest and decision trees classifier with small value of Mean-Squared-Error (MSE).

After analyzing different studies that have been undertaken by different researchers worldwide on malaria incidence cases prediction by using different machine learning algorithms as clarified in the above research studies, we find that those different studies generated different values of prediction accuracy and even the best model candidate varies from country to country. Sometimes, machine learning projects may require a huge amount and good quality of data for allowing the predictive model to make the best prediction with high accuracy [12]. This indicates that with different geographic locations and even the vegetation, habitation, social, economic, public health and political situation might impact the predictive accuracy [2].

The researchers adopted different evaluation metrics which may affect the resulted model performance in one way or another. For instance, the prediction accuracy evaluation metric alone tends to hide useful information about model performance on each class to be predicted . The researchers did not also address the way the predictive model has to be integrated with Internet of Things(IoT) data from field sensors for helping the general public to view real time information about malaria outbreak prediction based on environmental climate conditions via there mobile devices such as Smart-phone,table or Personal Computer(PC)which may lead to take further measures for reducing the spread of malaria transmission across the citizens.
An IoT is an interconnection of physical devices embedded with electronics and software, applications and virtual world for allowing different components of the system to communicate and exchange information via sensors connection and internet access, and finally providing data exchange with manufactures, operators and other connected devices [13–15].Internet of Things allows the physical objects embedded with sensors to be monitored and controlled remotely at any time and at any place by using existing network connectivity.

5 Research Methodology

For conducting this research activity, we have used different approaches.Thus, this section gives a brief explanations about research methodology approaches used by the researcher throughout in this study. These include Data Collection, Data Visualization, Data Preparation and Wrangling, Machine Learning models Training and Evaluation, Designing and implementation of Real-Time IoT based system, and finally integration of IoT system with Machine Learning predictive model for doing prediction on real time data from field sensors.

5.1 Data Preparation and Wrangling

Several categories of data used throughout this study were collected separately from different sources and were contained in different format structures. As show in the Fig.1 and Fig.2, the malaria data used contains 96 samples recorded by MINISANTE during of 96 months spans between 2012 and 2019 and aggregated on monthly basis. However, the temperature and humidity and rainfall data collected contained in samples recorded by METEO Rwanda based on daily basis from 2012 to 2019.

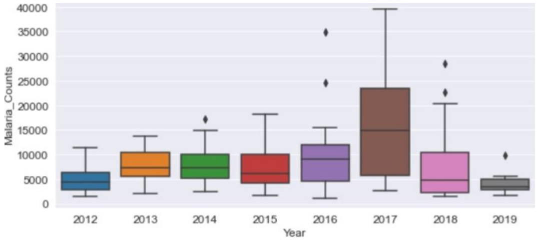


Fig. 1. Bugesera-Malaria Distribution

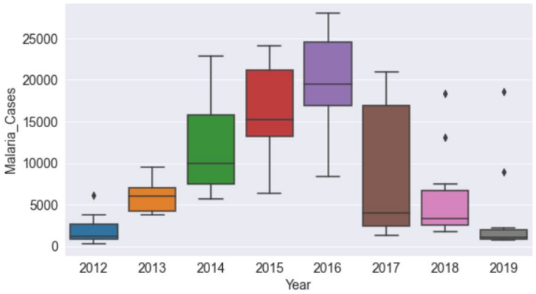


Fig. 2. Huye-Malaria Distribution

By using data science techniques, the temperature and humidity samples were averaged on monthly basis, and the rainfall samples were summed based on monthly basis because the climate season does not change a lot in single day or week. Therefore, we have assumed that the climate variables of interest can be changed in a remarkable way after one month.Finally, we come up with 96 samples of temperature, humidity and rainfall that span from 2012 to 2019.

Then after, the resulted separate datasets with 96 samples for each were combined together to construct a single dataset that contains features and target variables. Each malaria cases number were divided by the corresponded number of district population for obtaining the malaria transmission ratio. Among 96 samples of the temperature data and humidity data, there were 6 and 30 missing samples of temperature and humidity respectively. The missing values of temperature data were replaced by mean values of temperature in the specific months, while the rows with humidity missing data were removed for optimizing prediction accuracy.

For being able to classify if the malaria transmission rate is at Low- or High-level risk, the total values contained in malaria transmission ratio feature were divided into two categories as follows. Firstly, we subtracted minimum from maximum malaria transmission ratio and then divide the result by three to generate a constant(r) to be used for defining the categories of malaria transmission rate as follow [9]:

1. Category_1:
$$min_ratio, min_ratio + r$$
2. Category_2:
$$min_ratio + 2 * r, max_ratio$$

The Category_1 and Category_2 define the range of malaria transmission ratios associated with low and high risk of malaria transmission respectively. At the end we have added to the dataset another column named “class” whose values were “0” (Low level malaria transmission risk) and “1” (High level malaria transmission risk). “0” is assigned if the malaria transmission ratio falls in the Low Risk transmission level interval (Category_1). However, “1” is assigned if the malaria transmission ratio falls in the High Risk transmission level interval (Category_2).

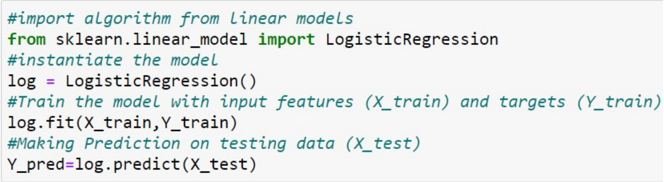
5.2 Predictive Models Modeling

This part gives a detailed explanation and implementation procedures of different Machine Learning Classification algorithm used by the researcher throughout this research work.The classification algorithms are the subset of machine learning supervised algorithms that are used to discover the categories of unknown observations [16]. These algorithms have been used for mapping the relationship between the dependent variables (model input) and independent variables (model target). They include Logistic Regression, Random Forest Classifier, K-Nearest Neighbors Classifier, Multi-Layer Perceptron Classifier, Decision Tree Classifier and Gradient Boosting Classifier. These algorithms were selected because they are popular in solving machine learning classification problems and availability of huge documentation because a large community is using them. The next paragraphs give a brief description about each one among these stated algorithms above.

a) Logistic Regression: A Logistic Regression is a type of machine learning supervised algorithm used to make prediction when the target variables are discrete or categorical and commonly known in solving binary classification problems such as spam detection, cancer detection, anomaly detection [16]. Unlike Linear regression which predicts unbound values, for Logistic Regression the range of predicted values is known.The mathematical expression of Logistic Regression is given by F(X) [17].

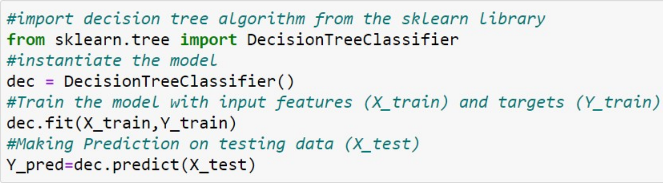
F(X) = sigmoid (WX+ b)

Here, X is the input feature vector. W, band sigmoid are the weight vector, bias and activation function respectively. Weight vector and bias are the model parameters to be identified during of model training. The sigmoid function or activation function is used for mapping the values between 1 and 0. If the output of the sigmoid function is above 0.5 we can classify this as 1 and 0 is the output is below 0.5 [18]. The following python codes show how Logistic Regression can be implemented.



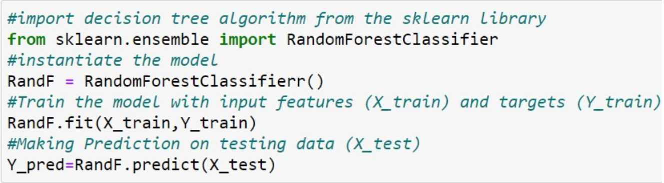
```
#import algorithm from linear models
from sklearn.linear_model import LogisticRegression
#instantiate the model
log = LogisticRegression()
#Train the model with input features (X_train) and targets (Y_train)
log.fit(X_train,Y_train)
#Making Prediction on testing data (X_test)
Y_pred=log.predict(X_test)
```

b) Decision Tree Classifier: Decision tree Classifier is a supervised machine learning algorithm used in machine learning and in statistics when the target variables are categorical. This predicting modelling approach uses a tree-like graph as a predictive model where observations are represented the branches and target values or the actual output or class represented in the leaves. The goal of this algorithm is to build a predictive model that can predicts the target value by learning decision rules identified from the features. These rules are implemented by using if-then-else statements [19]. The following python codes provide the implementation of this algorithm.



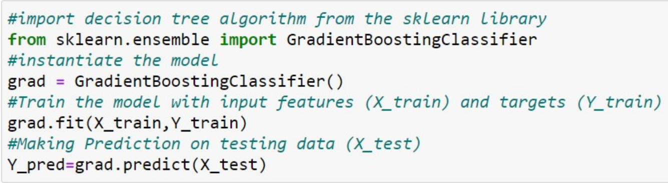
```
#import decision tree algorithm from the sklearn library
from sklearn.tree import DecisionTreeClassifier
#instantiate the model
dec = DecisionTreeClassifier()
#Train the model with input features (X_train) and targets (Y_train)
dec.fit(X_train,Y_train)
#Making Prediction on testing data (X_test)
Y_pred=dec.predict(X_test)
```

c) Random Forest Classifier: The decision tree machine learning can sometimes suffer from high variance, these may impact their results negatively to the specific training data. This variance can be reduced by building multiple predictive models in parallel from multiple samples of your training data, however these trees might be highly correlated and this can make the predictions to be similar. Random Forest algorithm is a supervised machine learning algorithm that uses multiple trees identified from the samples of your training data and forced them to be different by limiting the features that each model can evaluate for each sample. The final prediction is the class that comes many times in the output of the multiple trees used for the specific training data [20].The following program code shows how the algorithm was implemented.



```
#import decision tree algorithm from the sklearn library
from sklearn.ensemble import RandomForestClassifier
#instantiate the model
RandF = RandomForestClassifierr()
#Train the model with input features (X_train) and targets (Y_train)
RandF.fit(X_train,Y_train)
#Making Prediction on testing data (X_test)
Y_pred=RandF.predict(X_test)
```

d) Gradient Boosting Classifier: A gradient boosting classifier is one type of ensemble techniques used in machine learning for increasing the prediction accuracy. It involves a collection of the weak models to build a strong predictive model. Decision tree algorithms are usually used to build a gradient boosting classier. Gradient boosting classifier is used to make a prediction when the target variables are categorical [21]. The following python codes show how the algorithm was implemented using python built-in library.



```
#import decision tree algorithm from the sklearn library
from sklearn.ensemble import GradientBoostingClassifier
#instantiate the model
grad = GradientBoostingClassifier()
#Train the model with input features (X_train) and targets (Y_train)
grad.fit(X_train,Y_train)
#Making Prediction on testing data (X_test)
Y_pred=grad.predict(X_test)
```

e) K-Nearest Neighbors Classifier (KNN): The K-Nearest Neighbors is machine learning algorithm used in finding similarities between data. During the model training phase all of the data are used for learning the similarities between data. Then during of model prediction for unseen data, the model searches through the entire dataset the K-most similar training examples to new example and the data with K-most similar instance is returned as the prediction. The algorithm states that if you are similar to your neighbours, that means that you are one of them [22]. In K-Nearest Neighbours, K means the number of neighbor points which contribute in voting.

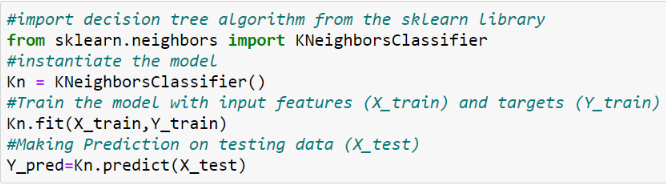
In KNN the voting points are selected by using Euclidean distance between the new point and the existing points and then the points with least distances are selected. The general formula of Euclidean distance is given by the following mathematical expression [23].

$$d(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}$$

Where,

p, q = two points in Euclidean n-space
q_i, p_i = Euclidean vectors, starting from the origin of the space (initial point)
n = n-space

The following python codes implement this algorithm via python library.

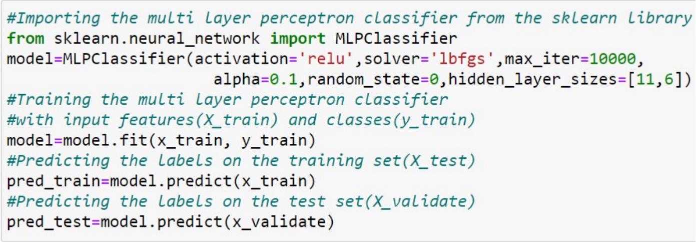


```
#import decision tree algorithm from the sklearn library
from sklearn.neighbors import KNeighborsClassifier
#instantiate the model
Kn = KNeighborsClassifier()
#Train the model with input features (X_train) and targets (Y_train)
Kn.fit(X_train,Y_train)
#Making Prediction on testing data (X_test)
Y_pred=Kn.predict(X_test)
```

f) Artificial Neural Network (ANN): The Artificial Neural Network is one of popular machine algorithms used for regression, classification problems and data processing. An Artificial Neural Network is mathematical algorithm implemented based on the architecture and the functionality of human biological neurons. The units of ANN are the neurons and these neurons are connected by using weighted links. During of model training he training data X are applied to a neuron as inputs and process them into outputs Y. The output of each neuron is computed by combining linearly the inputs data Xi , Weights Wi and the bias band then pass the result into a non-linear function(activation function) to produce the final output [24, 25] The following mathematical equation illustrates how the output Y of the neural network unit is computed. Where f denotes the activation function, bthe bias and the N means the number of inputs to the neuron [25].

$$\mathrm{Y} = f(\sum_{i=1}^{N} w_i x_i \ + \mathrm{b})$$

The neural network is constructed by combining multiple neuron unit together and stacking them together. As seen in the Fig.3, in the current research we have used only three layers for implementing the neural network architecture. These three layers include 2 hidden-layers with 11and 6 neurons in the first and second hidden layer respectively and one output neuron with one neuron. The activation function(f) was implemented by using Rectified-Linear-Unit(Relu) nonlinear function. The following program code shows how the algorithm was implemented via pythom programming language framework.



```
#Importing the multi layer perceptron classifier from the sklearn library
from sklearn.neural_network import MLPClassifier
model=MLPClassifier(activation='relu',solver='lbfgs',max_iter=10000,
                    alpha=0.1,random_state=0,hidden_layer_sizes=[11,6])
#Training the multi layer perceptron classifier
#with input features(X_train) and classes(y_train)
model=model.fit(x_train, y_train)
#Predicting the labels on the training set(X_test)
pred_train=model.predict(x_train)
#Predicting the labels on the test set(X_validate)
pred_test=model.predict(x_validate)
```

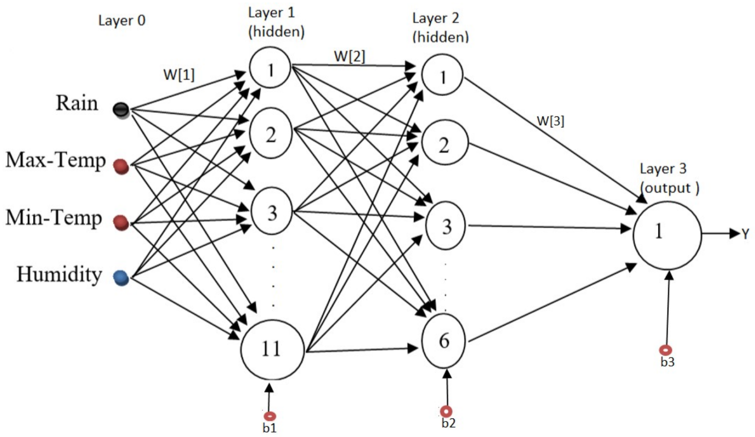


Fig. 3. Artificial Neural Network architecture

5.3 Model Training and Evaluation

a) Features Selection: The feature engineering is one of the core techniques that can be used to increase the chances of success in solving machine learning problems [26]. Before applying the whole features of the dataset as the input for the machine learning algorithm, Decision Tree machine learning algorithm was used to select the important features for predicting the malaria transmission rate. The main predictors considered during of training and evaluation processes include monthly minimum temperature,maximum temperature, rain fall intensity and relative humidity.

The Fig.4 indicates that the humidity,rainfall intensity and minimum temperature are the features of importance for training and could generate prediction with high accuracy.However, the maximum temperature is not important during this training and should not considered.

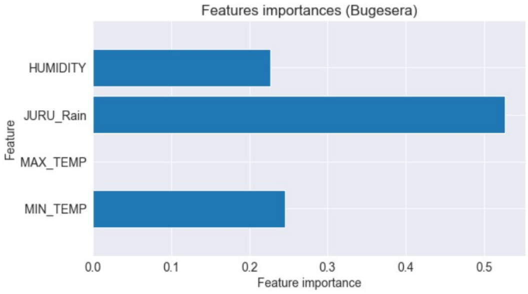


Fig. 4. Features Selection

b) Model training and Evaluation: The dataset used contains 96 samples that include four features and one target variable called class. As shown in the Fig.5, before model training the dataset was first split into two small datasets, training dataset and testing dataset. Each dataset among those small datasets generated contains four input features, one target variable and 96 samples. The training dataset contains 75% of original dataset, and the testing dataset contains 25% of the original dataset. The training dataset was applied to the machine learning algorithms for generative predictive model. However, the testing dataset was used for testing if the predictive model does not under-fitting or over-fitting [27].The overfitting is the situation where the predictive model predicts well on the training data but does not do the same for unseen data.

After preparing the dataset to be used by the algorithms, the models were trained on training dataset to model the relationship between the environmental data such temperature, humidity and rainfall and the transmission rate of malaria mosquitoes by using python programming concept. The python programming environment was selected because python has seem to be a stable and popular language that makes many tools available for the researcher in the whole process of an AI project [28–30].

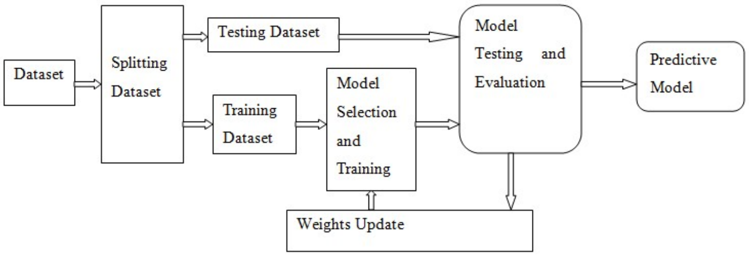


Fig. 5. Model training and Evaluation

Finally, we have evaluated the generated Machine Learning predictive models for selecting the best predictive model using predictive accuracy, precision and recall machine learning evaluation metrics for classification models. The prediction accuracy is used to measure how well the generated predictive model is performing. In other words, it compares a predicted value and an observed or known value. The higher prediction accuracy value indicates that the model is performing better [31, 32].

5.4 Design Real Time IoT based data collection system

The sensing part of the data collection system considered in this study contains three sensors for measuring environmental temperature, relative humidity and rain intensity and NodeMcu (ESP8266) as a microcontroller platform with Internet connection capability for transmitting the sensor data to the IoT cloud through existing Internet based network connectivity.

According to the Fig.6,the collected data were published with specific topic to MQTT Broker via IoT Gateway specifically network router or access point which is connected directly to internet connectivity. MQTT broker is one of IoT components that serves as the intermediate channel or path between two communicating devices or between devices and application platforms for allowing these devices and application to exchange data information at low power consumption [33, 34]. An IoT Gateway or a network Gateway is a device or a system which primarily allows the devices that traditionally having no internet connection capability to be connected [35, 36].

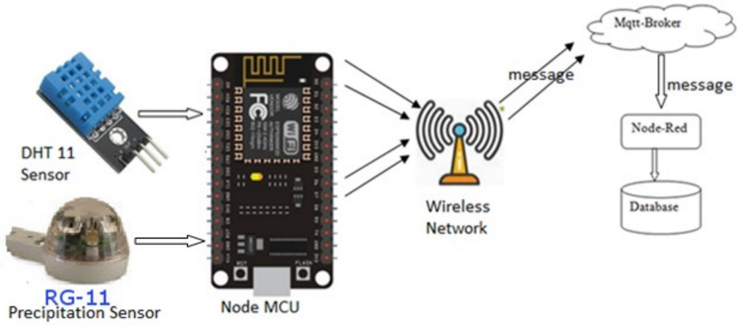


Fig. 6. IoT System Architecture

At the other side of the MQTT broker, the local server implemented by using Node-Red software tool, subscribed to the same topic and MQTT Broker as well as did by the publishing field sensor nodes.The Node-RED is a software tool that helps the engineers to develop prototypes and applications that can collect data and communicate data remotely or trigger on event such as IoT applications [37, 38].

The stream of IoT data are the type of big data that need to be managed: stored, analyzed in real time or later for uncovering new insights. Therefore, for storing and doing analytics on an incoming stream of IoT data, we have built a mongoDBdatabase for storing permanently data from field sensors. A mongoDBdatabase is a type of no-SQL database used for storing structured and unstructured data [39]. In the return the received data by the Node-Red data collector nodes by the help of MQTT IoT communication protocol are pushed into the database to be stored for the later use.
The Fig.7 shows how the communication between MQTT broker and database was implemented by using Node-Red software tool.The figure contains three main nodes: MQTT, function and mongoDbnodes. MQTT node is used to create active connection with MQTT broker,function node helps to select the specific topic from the multiple topics and mongoDBnode creates a connection with local mongoDBdatabse.

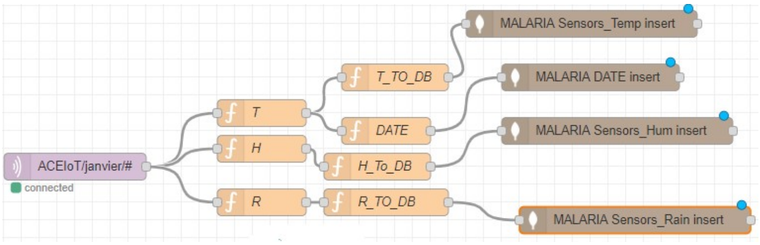

ACEIoT/janvier/#
connected
T
H
R
T_TO_DB
DATE
H_To_DB
R_TO_DB
MALARIA Sensors_Temp insert
MALARIA DATE insert
MALARIA Sensors_Hum insert
MALARIA Sensors_Rain insert

Fig. 7. Node-RED: MQTT Communication

Lastly as seen in the Fig.8, through python script the data in the database were collected into the format needed to be processed by the Machine Learning predictive model generated during of machine learning training process. Then after data preparation, the data were combined with the ML predictive model to generate new predictions and the resulted prediction sent to the node-red dashboard interface via local MQTT sever.

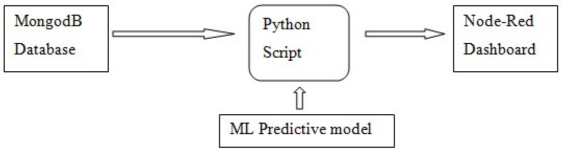
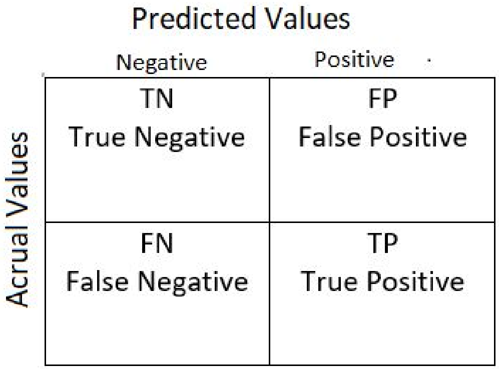


Fig. 8. System User Interface construction

6 Results and Discussions

6.1 Machine Learning Model Evaluation Metrics

The selection of the specific evaluation metrics depends on the type of category of machine learning problem. The metrics techniques used during of model evaluation in this research study includes Accuracy, Recall, Precision, sensitivity and specificity [32, 40]. These evaluation metrics are defined using confusion matrix and were selected because they are the best fit for machine learning classification problems. The confusion matrix is one of machine learning techniques for summarizing the performance of a machine learning classification models. As shown in the Fig.9 the matrix contains four main elements that include True Positive (TP) ,False Positive (FP) , True Negative and False Negative [31, 41].



	Predicted Values	
	Negative	Positive
Acrual Values	TN True Negative	FP False Positive
	FN False Negative	TP True Positive

Fig. 9. Confusion Matrix

Precision $= TP/(FP+TP)$, Recall $= TP(TP + FN)$

As seen in Fig.9, the True Positive and True Negative denote the number of positive and negative instances for binary classification that are correctly classified. However the False Positive and False Negative represent the number of positive and negative instances that are wrongly classified during of machine learning prediction [32]. These four elements are used for generating prediction accuracy, recall, precision and specificity.
The prediction accuracy tends to hide useful information about model performance. Therefore relying only on accuracy during machine learning evaluation, it can provide misleading information when prediction is made on unseen data [31]. So, for overcoming this problem others metrics such as recall, precision and specificity were used.

6.2 Results of Machine Learning Training and Evaluation process

The results appeared in the Fig.10 were generated by Multi-Layer Perceptron (MLP) and Logistic Regression (LR) machine learning algorithms during of model's training and evaluation process. As seen in the Fig.10, the predictive accuracy was taken as the machine learning evaluation metric to assess the performance of the predictive model. It is clear to figure out that the predictive accuracy is improved as the number of iterations increase progressively for both training and testing data.

However, when the number of iterations goes under of 150 and 5 for Multi-Layer Perceptron Classifier and Logistic Regression respectively, the model tends to underfit and overfit respectively. The term underfit means that the model performance is not good for both training and testing data, while the term overfit means that the model performance is very high and too low for training and testing data respectively. Thus, the best predictive model was taken after the model training process goes beyond of 150 iterations. Furthermore, the models tend to underfit for a lower value of iterations. The MLP Classifier generated 93.9% and 88.2% of training and testing accuracy respectively. However, the Logistic Regression generated 73.5% and 70.6% of training and testing accuracy respectively.

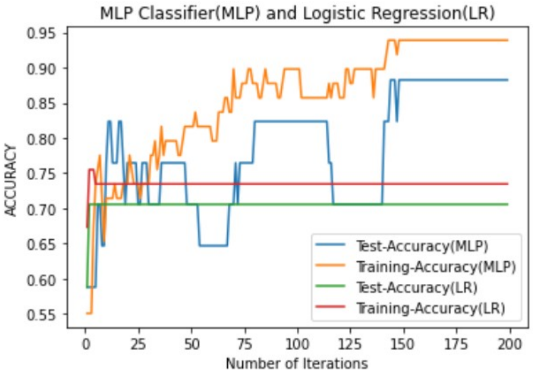


Fig. 10. MLP (MLP) Model and Logistic Regression (LR) Model Evaluation

The Fig.11 present the results of training and evaluation process for Random Forest (RF) and Gradient Boosting (GB) Classifiers. As shown by the Fig.11, the predictive accuracy was improved by increasing the number of estimators. However, the Random Forest and the Gradient Boosting predictive models tends to overfit as the number of estimators goes beyond of 20 and 30 respectively and tend to underfit for lower values. Thus, the Gradient boosting and Random Forest Classifiers were train on 20 and 15 estimators respectively. Finally, both algorithms reached the 98% and 82.4% of training and testing accuracy respectively.

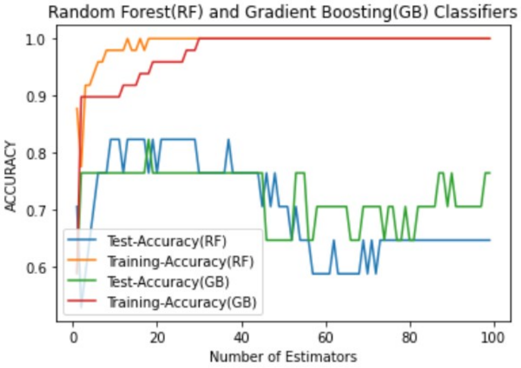


Fig. 11. Random Forest (RF) and Gradient Boosting (GB) Models Evaluation

The results of machine learning training and evaluation process of K-Nearest Neighbors and Decision Tree are shown in Fig.12 and Fig.13 respectively.The results show that the K-Nearest Neighbors tends to underfit comparing with the other algorithms; however, the decision tree tends to overfit for higher values of Max-Depth algorithm parameter. The decision tree predictive model was selected when the Max-Depth equals to 3, while the predictive model of the K-Nearest Neighbors was selected when the number of neighbors equals to 3 for avoiding the overfiting problems.

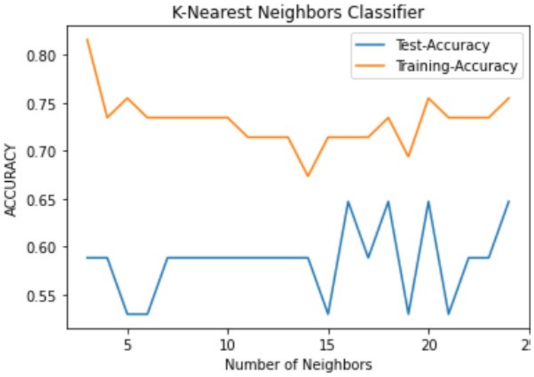


Fig. 12. KNN Model Evaluation

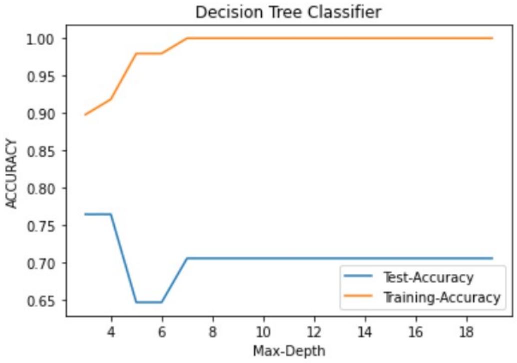


Fig. 13. Decision Tree Model Evaluation

Moreover, apart from machine learning predictive accuracy, precision and recall model performance metrics were investigated to evaluate how well the generated predictive model is performing and comparing the results generated by different machine learning algorithms used. Thus, the figures 14,15,16 and 17 summarize the results of model training and evaluation process from Bugesera and Huye district.

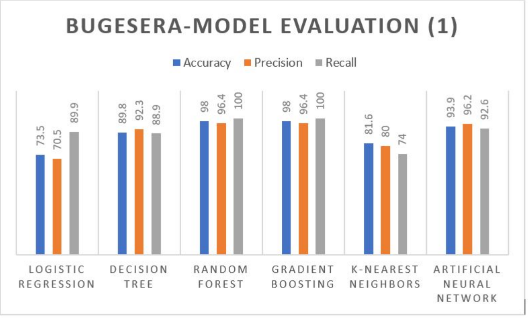


Fig. 14. Models-Evaluation(Bugesera)(a)

According to the results shown in figures 14,15,16 and 17 that describe the models evaluation process, it is seen that using Artificial Neural Network, Random Forest algorithms and Gradient Boosting machine learning algorithms could perform well respectively in both training and testing prediction accuracies comparing with the other algorithms for the current research.
The prediction accuracy and precision metrics of Huye district reported are very small comparing with the results seen in the Bugesera district during of model evaluation process. This may be justified by the dataset used in Huye district which didn't contain relative humidity data in both training and testing process.

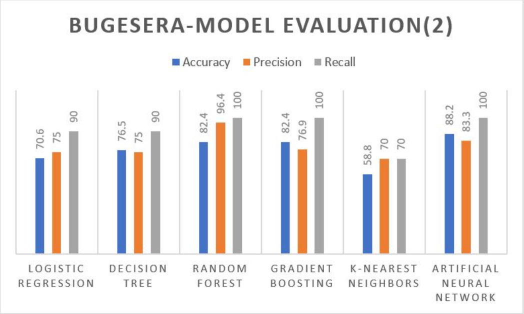


Fig. 15. Models-Evaluation(Bugesera)(b)

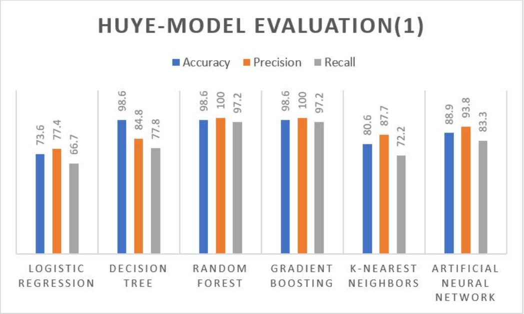


Fig. 16. Models-Evaluation(Huye)(a)

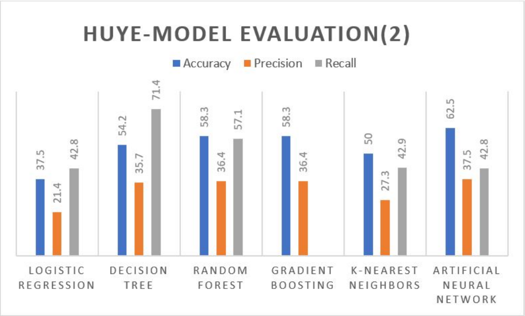


Fig. 17. Models-Evaluation(Huye)(b)

6.3 System dashboard/User interface

After machine learning training and evaluation process, the Artificial Neural Network predictive model was integrated with the IoT based real time data collection system using python script to make prediction on future data from field sensors. Fig.18 shows how the data from field sensors could be accessed through end devices such as tablet, Smartphone or PC via internet connection. The above dashborad displays the real time temperature, humidity and rain measurements accessed from remote sensors nodes.

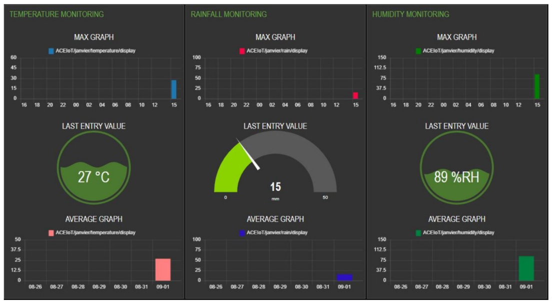


Fig. 18. Sensors-Measurement Dashboard

The Fig.19 shows how the results of machine learning analytics (prediction on sensors data) could be accessed through tablet, Smartphone or PC. Through the following figure of prediction dashboard, the system generates "Normal" as prediction. This means that the malaria transmission rate is at normal level at the indicated date.

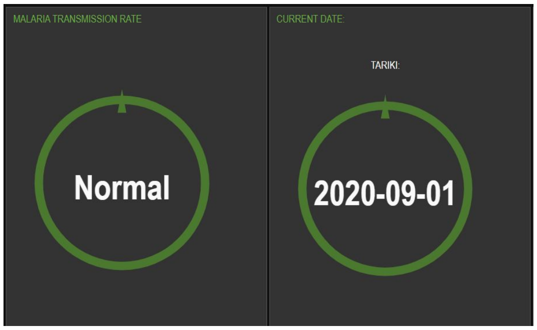


Fig. 19. Malaria-Prediction-User-Interface

7 CONLUSIONS AND FUTURE WORK

This research aimed to map the dependency between malaria transmission outbreak and climate environmental variables such as temperature, relative humidity and rain fall in Rwanda by using Machine Learning and Internet of Things. The outcome of the current research shows that the malaria transmission rate could be predicted well by using Artificial Neural Network (ANN) and Random Forest Machine Learning Algorithms respectively comparing with the other algorithms tested through this research. During the model performance evaluation,93.9% and 88.2% of training and testing prediction accuracies respectively were achieved by using ANN in Bugesera district. However, 88.9% and 62.5% of training and testing prediction accuracies were generated by using the same predictive model in Huye district because of lack some predictors like relative humidity.

The models used in this current study, were trained and evaluated by using 6 years(2012-2019) climate, population and malaria data from Rwanda. However, the government malaria prevention policies like use of mosquito nets and use of pesticides inside and outside the house to kill mosquitoes can impact the transmission rate of the malaria disease.

For the future work, we expect to employ some advanced machine learning models like time series models(Recurrent Neural Networks) to predict the future behavior of malaria transmission. Secondary, the number of observations will be increased and the government input policies to prevent the transmission of malaria disease will be considered during the data analysis.

References

1. M. GASANA et al., “How can we overcome Malaria threat and make a Rwanda free of Malaria ?” 2017. H.[Online]. Available: https://www.afro.who.int/news/how-can-we-overcome-malaria-threat-and-make-rwanda-free-malaria. [Accessed: 03-Mar-2020].
2. R. Kiang et al., “Meteorological, environmental remote sensing and neural network analysis of the epidemiology of malaria transmission in Thailand.,” Geospat. Health, vol. 1, no. 1, pp. 71–84, 2006, doi: 10.4081/gh.2006.282.
3. D. D. Gashumba, “Annual Health Statistics Booklet of 2016,” 2016.
4. Hakizimana, “Malaria Operational Plan FY 2018,” vol. Malaria Jo, p. 72, 2018.
5. A. S. Walsh et al., “Predicting seasonal abundance of mosquitoes based on off-season meteorological conditions,” December, 2007.
6. S. Thakur and R. Dharavath, “Artificial neural network based prediction of malaria abundances using big data: A knowledge capturing approach,” Clin. Epidemiol. Glob. Heal., vol. 7, no. 1, pp. 121–126, 2019, doi: 10.1016/j.cegh.2018.03.001.
7. N. O. Adeboye, O. V. Abimbola, and S. O. Folorunso, “Malaria patients in Nigeria: Data exploration approach,” Data Br., vol. 28, p. 104997, 2020, doi: 10.1016/j.dib.2019.104997.
8. B. E. Chekol and H. Hagras, “Employing Machine Learning Techniques for the Malaria Epidemic Prediction in Ethiopia,” 2018 10th Comput. Sci. Electron. Eng. Conf. CEEC 2018 - Proc., pp. 89–94, 2019, doi: 10.1109/CEEC.2018.8674210.

9. G. Kalipe, V. Gautham, and R. K. Behera, “Predicting Malarial Outbreak using Machine Learning and Deep Learning Approach: A Review and Analysis,” Proc. - 2018 Int. Conf. Inf. Technol. ICIT 2018, pp. 33–38, 2018, doi: 10.1109/ICIT.2018.00019.
10. “World malaria report 2019,” World Health Organization. [Online]. Available: https://www.who.int/news-room/feature-stories/detail/world-malaria-report-2019. [Accessed: 20-Mar-2020].
11. O. P. Zacarias and H. Bostrom, “Comparing support vector regression and random forests for predicting malaria incidence in Mozambique,” Int. Conf. Adv. ICT Emerg. Reg. ICTer 2013 - Conf. Proc., pp. 217–221, 2013, doi: 10.1109/ICTer.2013.6761181.
12. Y. Roh, G. Heo, and S. E. Whang, “A Survey on Data Collection for Machine Learning: A Big Data - AI Integration Perspective,” IEEE Trans. Knowl. Data Eng., pp. 1–1, 2019, doi: 10.1109/tkde.2019.2946162.
13. F. Xia, L. T. Yang, L. Wang, and A. Vinel, “Internet of Things,” pp. 1101–1102, 2012, doi: 10.1002/dac.
14. I. Y, E. A, I. A, and T. H, “Internet of Things Architecture: Recent Advances, Taxonomy, Requirements, and Open Challenges,” no. June, pp. 10–16, 2017.
15. A. Gael, L. Filip, and F. Dragan, “IoThings: A Platform for Building up the Internet of Things.” Springer International Publishing AG 2018 V.E. Balas et al. (eds.), Soft Computing Applications, Advances in Intelligent Systems and Computing 633, DOI 10.1007/978-3-319-62521-8_15
16. A. M. Coroiu, “Model Evaluation as Approach to Predict a Diagnosis,” Springer International Publishing AG 2018 V.E. Balas et al. (eds.), Soft Computing Applications, Advances in Intelligent Systems and Computing 634, DOI 10.1007/978-3-319-62524-9_1
17. M. Stojiljković, “Logistic Regression in Python,” 2020. [Online]. Available: https://realpython.com/logistic-regression-python/. [Accessed: 04-Nov-2020].
18. A. Navlani, “Understanding Logistic Regression in Python,” 2019. [Online]. Available: https://www.datacamp.com/community/tutorials/understanding-logistic-regression-python. [Accessed: 04-Nov-2020].
19. Shubham, “Decision Tree,” 2020. [Online]. Available: https://www.hackerearth.com/practice/machine-learning/machine-learning-algorithms/ml-decision-tree/tutorial/. [Accessed: 04-Nov-2020].
20. W. Lin, Z. Wu, L. Lin, A. Wen, and J. Li, “An ensemble random forest algorithm for insurance big data analysis,” IEEE Access, vol. 5, no. JULY, pp. 16568–16575, 2017, doi: 10.1109/ACCESS.2017.2738069.
21. D. Nelson, “Gradient Boosting Classifiers in Python with Scikit-Learn,” 2020. [Online]. Available: https://stackabuse.com/gradient-boosting-classifiers-in-python-with-scikit-learn/. [Accessed: 04-Nov-2020].
22. J. Brownlee, “Develop k-Nearest Neighbors in Python From Scratch,” 2019. [Online]. Available: https://machinelearningmastery.com/tutorial-to-implement-k-nearest-neighbors-in-python-from-scratch/. [Accessed: 04-Nov-2020].
23. A. Robinson, “How to Calculate Euclidean Distance,” 2020. [Online]. Available: https://sciencing.com/how-to-calculate-euclidean-distance-12751761.html. [Accessed: 04-Nov-2020].
24. E. Manitsas, R. Singh, B. C. Pal, and S. Member, “An Artificial Neural Network Approach for Pseudo Measurement Modeling,” vol. 27, no. 4, pp. 1888–1896, 2012.
25. D. V. Coury and D. C. Jorge, “Artificial neural network approach to distance protection of transmission lines,” IEEE Trans. Power Deliv., vol. 13, no. 1, pp. 102–108, 1998, doi: 10.1109/61.660861.

26. J. Brownlee, “How to Choose a Feature Selection Method For Machine Learning,” 2019. [Online]. Available: https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/.
27. H. K. Jabbar and R. Z. Khan, “Methods to Avoid Over-Fitting and Under-Fitting in Supervised Machine Learning (Comparative Study),” no. December 2014, pp. 163–172, 2015, doi: 10.3850/978-981-09-5247-1-017.
28. Prince Patel, “Why Python is the most popular language used for Machine Learning,” 2018. [Online]. Available: https://medium.com/@UdacityINDIA/why-use-python-for-machine-learning-e4b0b4457a77. [Accessed: 20-Mar-2020].
29. N. Gupta, “Why is Python Used for Machine Learning?,” 2019. [Online]. Available: https://hackernoon.com/why-python-used-for-machine-learning-u13f922ug. [Accessed: 20-Mar-2020].
30. A. Beklemysheva, “Why Use Python for AI and Machine Learning.” [Online]. Available: https://steelkiwi.com/blog/python-for-ai-and-machine-learning/. [Accessed: 20-Mar-2020].
31. N. S. Chauhan, “Model Evaluation Metrics in Machine Learning,” 2020.
32. H. M and S. M.N, “A Review on Evaluation Metrics for Data Classification Evaluations,” Int. J. Data Min. Knowl. Manag. Process, vol. 5, no. 2, pp. 01–11, 2015, doi: 10.5121/ijdkp.2015.5201.
33. P. Jutadhamakorn, T. Pillavas, V. Visoottiviseth, R. Takano, J. Haga, and D. Kobayashi, “A scalable and low-cost MQTT broker clustering system,” Proceeding 2017 2nd Int. Conf. Inf. Technol. INCIT 2017, vol. 2018-Janua, pp. 1–5, 2017, doi: 10.1109/INCIT.2017.8257870.
34. U. Hunkeler, H. L. Truong, and A. Stanford-clark, “MQTT-S – A Publish / Subscribe Protocol For Wireless Sensor Networks.”
35. H. Chen, X. Jia, and H. Li, “A brief introduction to iot gateway,” IET Conf. Publ., vol. 2011, no. 586 CP, pp. 610–613, 2012, doi: 10.1049/cp.2011.0740.
36. S. Guoqiang, C. Yanming, Z. Chao, and Z. Yanxu, “Design and implementation of a smart IoT gateway,” Proc. - 2013 IEEE Int. Conf. Green Comput. Commun. IEEE Internet Things IEEE Cyber, Phys. Soc. Comput. GreenCom-iThings-CPSCom 2013, pp. 720–723, 2013, doi: 10.1109/GreenCom-iThings-CPSCom.2013.130.
37. A. Rajalakshmi and H. Shahnasser, “Internet of Things using Node-Red and Alexa,” pp. 3–6, 2017.
38. M. Leki and G. Gardaševi, “IoT sensor integration to Node-RED platform,” no. March, pp. 21–23, 2018.
39. Z. Wei-Ping, L. Ming-Xin, and C. Huan, “Using MongoDBto implement textbook management system instead of MySQL,” 2011 IEEE 3rd Int. Conf. Commun. Softw. Networks, ICCSN 2011, pp. 303–305, 2011, doi: 10.1109/ICCSN.2011.6013720.
40. P. A. Flach, “The Geometry of ROC Space: Understanding Machine Learning Metrics through ROC Isometrics,” Proceedings, Twent. Int. Conf. Mach. Learn., vol. 1, pp. 194–201, 2003.
41. J. Brownlee, “What is a Confusion Matrix in Machine Learning,” 2020.