Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 10, October-2022

Covid-19 Prediction based on Symptoms using Machine Learning

Arushi, Sunita Jalal, Chetan Singh Negi
College of Technology, Pantnagar

***Abstract*:- COVID-19 started in the Chinese province of Hubei's Wuhan in December 2019. Since then, several waves of covid-19 have hit people all around the world. As the whole world was striving to combat the coronavirus disease (COVID-19), healthcare and health monitoring systems were struggling to confront the virus. Many cases had been observed where COVID-19 could not be identified at a specific time. Furthermore, any effective strategy that could monitor the coronavirus state in the human body had not been established. As a result, patients with the coronavirus could not receive proper treatment when necessary. Therefore, the death toll due to COVID-19 was rising. Although the situation of covid-19 has subsided currently, precautions need to be taken in advance to keep further waves at bay. This paperwork proposes a systematic approach to combat the COVID-19 pandemic more efficiently by using various machine learning algorithms and comparing their accuracy and using the best outcome to predict the disease. With eight binary features, the model was able to predict the COVID-19 test outcomes with high accuracy. This paper suggests a practical solution with the help of the developed health monitoring system that can mitigate the loss done by COVID-19. When testing resources are few, this framework can be used, among other things, to prioritize testing for COVID-19.**

Keywords:-COVID-19 · Machine learning · Open source dataset · Data preprocessing · Confusion matrix

1 INTRODUCTION

Since the declaration of the widespread virus outbreak, a pandemic in March 2020 by WHO (World Health Organization) and the imposition of lockdown in most nations, many concerns have been raised over the upsurge of different waves and variants from time to time. The danger of covid-19 has not abated yet. As per the reports in the Times of India on 20 June 2022, the Indian Central Government had raised concerns over low testing and vaccination in nine states that were witnessing an upsurge in covid-19 cases and test positivity rates. According to Health secretary Rajesh Bhushan in a high-level review meeting with the states "Any laxity will result in deterioration of the situation in these districts".

The virus has hit 213 countries as of the third week of June 2022, infecting about 545,788,600 people (with active cases exceeding 17,758,752) and leading to a death toll of over 6,343,588 persons. Besides this, thanks to the limited resources available and the increased burden on the healthcare staff, patients infected with other diseases experienced delayed treatments. All these factors once again made us question even the nations with the best healthcare services around the world.

The impact on healthcare systems is often reduced by effective screening, which enables early and accurate identification of COVID-19. To support medical professionals around the world in prioritizing patients, particularly within the context of constrained healthcare resources, prediction models that incorporate many variables to work out the likelihood of infection have been developed. This study presents different machine-learning models that, by posing eight straightforward questions, can predict whether a SARS-CoV-2 RT-PCR test would be positive or not. The models use machine learning algorithms like gradient boosting, random forest classifier, logistic regression, KNN classification, Naive Bayes and support vector machine and clinical symptoms and integration of those features. The models were trained on data of all individuals in Israel tested for SARS-CoV-2 (70% of the total dataset) and tested on both the Israeli test dataset (30% of the entire dataset)[3] and data collected through the survey done by our team[4].

We have performed a comparative analysis on the six models trained and chosen the best one giving the results with high accuracy. Thus, selected model can be implemented at a good scale for effective screening and prioritization of testing for the virus in the general population.

To make the system user-friendly we have used android as an interface using kotlin. This is a fully functional application which asks registered users 8 questions and predicts the result using our integrated machine learning model. Figure 1 shows the modular structure of our application together with the overview of the backend and how users can interact with the app and get the predictions.

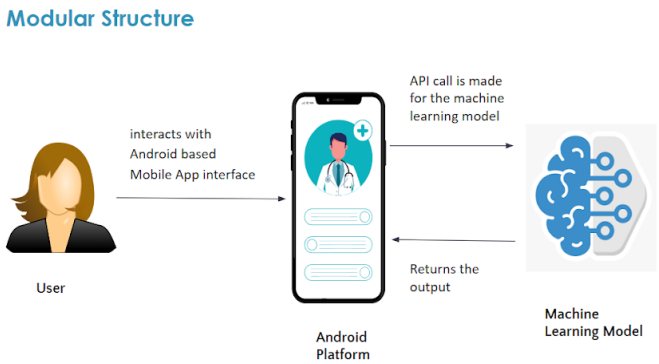

Figure 1. Modular Structure of the proposed system

2 RELATED WORKS

The human body is guarded by the immune system, but sometimes this system alone is not capable of preventing our body from diseases. Environmental conditions and living habits of individuals are the cause of many diseases

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 10, October-2022

that are the main reasons for a huge number of deaths in the world, and diagnosing these diseases sometimes becomes challenging. We need accurate, feasible, reliable, and robust systems to diagnose diseases in time so that these can be properly treated. With the expansion of medical data, many researchers are using these medical data and a few machine learning algorithms to help the healthcare communities in the diagnosis of many diseases. Numerous studies have been done related to predicting the disease using different machine learning techniques and algorithms which can be used by medical institutions. Continuous growth in medical data gave us how to extract the required information to predict the disease. Health data collected from patients can be used to predict various diseases with the help of modern techniques of data science and big data. These disease prediction models are vital to knowing the presence of disease. Various machine learning techniques like supervised, semi-supervised, unsupervised learning, etc. and raw medical data are required to detect different diseases. This data could easily be obtained from famous government hospitals. Machine learning techniques can use the data for the learning process and based on that learning they can predict the disease later. There are many literature reviews available in Disease Prediction. A study from Infectious Diseases of Poverty shows that machine learning techniques are often used to predict the severity of COVID-19, thereby enabling providers to optimize care. [1] AI and machine learning are often used to examine a person for COVID-19 as an alternative to traditional time-consuming and expensive methods. Although there are several studies on COVID-19, this study concentrated on the use of machine learning in forecasting COVID-19 cases and diagnosing patients for COVID-19 infection through their symptoms. Various machine learning models and apps have been made in the field of covid-19 prediction.

Mikko Vihtakari (May 2020)[6] made an app licensed by MIT License that illustrates how COVID-19 infection could develop in your country and why the drastic measures to fight the outbreak are justified. The model uses simple exponential math, and median estimates and ignores an entire lot of important parameters, like reporting error, development of immunity, population density, demography, variation, and uncertainty. Consequently, the model isn't accurate but gives an idea of how the outbreak could develop during the uncontrolled start phase most European countries have been going through in March 2020. The model parameters are adjusted for the situation in Norway 2020-03-17.

In [7], the authors developed App-based COVID-19 syndromic surveillance and prediction of hospital admissions: The COVID Symptom Study Sweden. The app-based COVID Symptom Study was launched in Sweden in April 2020 to contribute to real-time COVID-19 surveillance. Data from 19,161 self-reported PCR tests were wont to create a symptom-based model to estimate the individual probability of symptomatic COVID-19, with an AUC of 0.78 (95% CI 0.74–0.83) in an external dataset. The individual probabilities were wont to estimate daily regional COVID-19 prevalence, which was successively used together with current hospital data to predict next week's COVID-19 hospital admissions. It had been found that this hospital prediction model demonstrated a lower median absolute percentage error (MdAPE: 25.9%) across the five most populated regions in Sweden during the primary pandemic wave than a model based on case notifications (MdAPE: 30.3%). Identical error rates were found during the second wave.

Ramesh Kumar Mojjada, Arvind Yadav, A.V. Prabhu and Yuvaraj Natarajanc (2020)[8] made a study showing the power to predict the number of individuals who are affected by COVID-19 as a potential threat to human beings by ML modelling. During this analysis, the danger factors of COVID-19 were exponential smoothing (ES). The Lower Absolute Reductor and Selection Operator, (LASSo), Vector Assistance (SVM), and four normal potential forecasts, like Linear Regression (LR)). Each of those machine-learning models has three distinct kinds of predictions: the number of newly infected COVID 19 people, mortality rates and therefore the recovered COVID-19 estimates in the next 10 days.

Wie Kiang H. (2020)[9] in his article focussed on how machine learning is often used to study the spread of covid-19. The dataset was retrieved from the official repository of Johns Hopkins University. This data consists of daily case reports and daily statistic summary tables. Within the study, they need selected time-series summary tables in CSV format having three tables for confirmed, death, and recovered cases of COVID-19 with six properties. In concurrent, state-of-the-art mathematical models were chosen to support machine learning for a computational process to predict the spread of the virus, for instance: Support Vector Regression (SVR), Polynomial Regression (PR) and Deep Learning regression models. It also involved Artificial Neural Network (ANN) and Recurrent Neural Networks (RNN) using Long STM (LSTM) cells.

Celestine Iwendi, Ali Kashif Bashir, Atharva Peshkar, R. Sujatha, JyotirMoy Chatterjee, Swetha Pasupuleti, Rishita Mishra, Sofia Pillai and Ohyun Jo (July 2020) [10] worked on COVID-19 Patient Health Prediction Using Boosted Random Forest Algorithm. They used boosted random forest for prediction. Boosted Random Forest is an algorithm, which consists of two parts; the boosting algorithm: AdaBoost and therefore the Random Forest classifier algorithm, which successively consists of multiple decision trees. The model uses the COVID-19 patient's geographical, travel, health, and demographic data to predict the severity of the case and therefore the possible outcome, recovery, or death. The model gave results with an accuracy of 94% and an F1 score of 0.86 on the dataset used. It was observed that the patients' gender and deaths are positively correlated and that the bulk of patients is aged between 20 and 70 years.

M. Shobana, S. Vaishnavi, C. Gokul Prasad, P. Poonkodi, R. Sabitha, and S. Karthik (2022)[12] published a piece of

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 10, October-2022

writing in which they worked on 'Relating Design Thinking Framework in Predicting the Spread of COVID in Tamilnadu Using ARIMA'. ARIMA (Auto Regressive Integrated Moving Average) models are often used in forecasting the spread of COVID with the previous datasets extracted from Kaggle. They have considered the info from March 2020 to June 2021 and predicted the COVID cases for the next one month July 2021. In specific, it's concentrated in one particular state Tamilnadu from INDIA.

Ismail Kirbas, A. Sözen, Azim Doğuş Tuncer, F. Ş. Kazancıoğlu (June 2020)[13] performed a Comparative analysis and forecasting of COVID-19 cases in various European countries with ARIMA, NARNN and LSTM approach. during this study, confirmed COVID-19 cases in Denmark, Belgium, Germany, France, UK, Finland, Switzerland and Turkey were modelled with Auto-Regressive Integrated Moving Average (ARIMA), Nonlinear Autoregression Neural Network (NARNN) and Long-Short Term Memory (LSTM) approach. Six model performance metrics were wont to select the most accurate model (MSE, PSNR, RMSE, NRMSE, MAPE and SMAPE). consistent with the results of the first step of the study, LSTM was found to be the foremost accurate model. Within the second stage of the study, the LSTM model was provided to form predictions in a 14-day perspective that is yet to be known. Results of the second step of the study show that the entire cumulative case increase rate is expected to decrease slightly in many countries.

Yasminah Alali, Fouzi Harrou & Ying Sun (February 2022)[14] develop an assumption-free data-driven model to accurately forecast the COVID-19 spread. They started with Bayesian optimization to tune the Gaussian process regression (GPR) hyperparameters to develop a GPR-based model to forecast the recovered and confirmed COVID-19 cases in two highly impacted countries, India and Brazil. However, machine learning models don't consider the time dependency in the COVID-19 data series. Here, dynamic information has been taken under consideration to alleviate this limitation by introducing lagged measurements in constructing the investigated machine learning models. They also assessed the contribution of the incorporated features to the COVID-19 prediction using the Random Forest algorithm. Their results highlighted the superior performance of the dynamic GPR compared to the opposite models (i.e., Support vector regression, Boosted trees, Bagged trees, Decision tree, Random Forest, and XGBoost) and procured an averaged mean absolute percentage error of around

0.1%. They provided the arrogance level of the predicted results based on the dynamic GPR model and showed that the predictions are within the 95% confidence interval.

3 METHODOLOGY

Research methodology can be defined as the specific procedures or techniques used to identify, select, process, and analyze information about a few topics. [2] During a research paper, the methodology section allows the reader to critically evaluate a study's overall validity and reliability. The methodology section answers two main questions: How was the info collected or generated? How was it analyzed?

3.1 Data Processing

This consists of two steps, i.e., Data Collection and Data Pre-processing. Data can be referred to as the raw material. Therefore, the first step in the development of COVID-19 applications is data collection. Multiple datasets are put online in regards to COVID-19. Most if not all of those datasets are open source meaning that they are free for anyone to download and use. The dataset that we are using is formed using 8 attributes that were noted in 278848 Israeli patients and was gathered by the Israeli Ministry of Health. The dataset contains initial records, on a day to day basis, of all the residents who were tested for COVID-19 nationwide. The dataset's attributes include cough, fever, pharyngitis, shortness of breath, headache, corona result, age 60 and above, gender, and test indication. The corona result tells whether or not people may have the coronavirus in their bodies. The dataset's majority of variables are in binary format. If a feature variable's value is "1," it signifies that a specific symptom is present; if it is "0," there's no symptom. The symptoms taken are supported by guidelines given by the World Health Organization (WHO) and the Ministry of Health and Family Welfare, India.
The following attributes describe each of the dataset's features used by the model:

1. Gender- This feature is assessed into three categories- male, female and none. After data pre-processing, the specific data has been converted to numerical data. All the male categories are replaced with '0.0' and feminine ones with '1.0'. The third category, i.e., none has been replaced with the amount '3.0'.
2. Cough-This feature is assessed into two categories, presence and absence.'0' indicates the absence of cough and '1' indicates the presence of cough. After data pre-processing '0' is replaced with '0.0' and '1' is replaced with '1.0'.
3. Fever--This feature is assessed into two categories, presence and absence of fever .'0' indicates the absence of fever and '1' indicates the presence of

fever. After data pre-processing, '0' is replaced with '0.0' and '1' is replaced with '1.0'.

4. Sore_throat-This feature is assessed into two categories, presence and absence of sore_throat .'0' indicate the absence of sore_throat and '1' indicates the presence of sore_throat. After data pre-processing, '0' is replaced with '0.0' and '1' is replaced with '1.0'.
5. Shortness_of_breath -This feature is assessed into two categories, presence and absence of shortness_of_breath.'0' indicates the absence of it and '1' indicates presence. After data preprocessing, '0' is replaced with '0.0' and '1' is replaced with '1.0'.
6. head_ache -This feature is assessed into two categories, presence and absence of head_ache.'0' indicates the absence of head_ache and '1' indicates the presence of

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 10, October-2022

head_ache. After data preprocessing, '0' is replaced with '0.0' and '1' is replaced with '1.0'.

7. Age_60_and_above -This feature is assessed into three categories -None No, Yes. No indicates that the age is below 60 and Yes indicates that the age is above 60.

8. test_indication-This feature is assessed into three categories-Other, Abroad, Contact with Confirmed. This indicates whether an individual has come into contact with covid positive, he/she has come from abroad or there's any other reason for existing symptoms

9. Corona_result-This feature is assessed into three categories -Positive, Negative, and Other. Positive indicates that the individual is covid positive, negative indicates that the individual isn't covid positive, other indicates that there's no surety about the result it can be some other allergy also.

We have also used the data collected through a survey to test the models on the data of different regions and check the reliability of the model if it works the same on the data after two years of pandemic. In May 2022, the data was collected through a survey done on students of Govind Ballabh Pant University of Agriculture and Technology located in the district of Udham Singh Nagar, Pantnagar, Uttarakhand. Data consisted of 149 rows and with similar features to the other dataset. The features hold similar meaning as described for Israeli dataset.

Data preprocessing can be defined as a process of preparing the raw data and making it suitable for a

machine learning model. it's the first and crucial step while creating a machine learning model.

Steps that were followed during data pre-processing are:

1. Getting the Dataset

To create a machine learning model, we require a dataset as a machine learning model completely works on data. The data that was used to train and test the machine learning models was retrieved from the website of the Israeli Ministry of Health. Furthermore, the machine learning models were tested on another set of data collected through a survey done on the Indian population.

2. Importing the libraries

Various predefined python libraries were used for data pre-processing. Some of the libraries used are Numpy, Pandas, Seaborn, Sklearn, pickle etc.

3. Importing the Datasets

For performing on datasets collected for machine learning models, the present directory was set to the working directory. Then the datasets were imported. To import the dataset, the read_csv() function of the pandas library was used, which may read a CSV file and perform various operations on it. With this function, CSV files are often read both locally as well as through URL

4. Handling Missing data

If the dataset contains some missing data, then it's going to create a huge problem for our machine learning model. Therefore, it's required to handle missing values present in the dataset. The next step that was followed in data pre-processing was handling the missing data. The process that was followed was deleting the rows or columns having null values. If columns have quite half of the rows as null, then the entire column can be dropped. The rows which are having one or more column values as null also can be dropped.

5. ENCODING CATEGORICAL DATA

If there are categorical variables, it can cause trouble in building the model because the machine learning model completely works on mathematics and numbers. Therefore, the specific variables were encoded into numbers using replace function. The categorical variables were converted to the following numerical values

"No" to value=0.0
"Yes" to value=1.0
"0"to value=0.0
"1", value=1.0
"Male" to value=0.0
"Female" to value=1.0

"Other" to value=1.0
"Abroad", value=2.0
"Contact with confirmed" to value=3.0
"other "to value 2.0

6. Splitting the Dataset into the Training set and Test set [15]

Splitting the dataset into a training and test set is important because if we train the models on a certain dataset and test the models on a completely different dataset then it will be difficult for our model to understand the correlations between the models. If we train our model alright and its training accuracy is also very high, but we offer a new dataset to it, then it'll decrease the performance. So we always attempt to make a machine learning model which performs well with the training set and also with the test dataset.

The Israeli dataset[3] was divided into 70% of the dataset as the training set and 30% as the test set. Training Set can be described as a subset of dataset to coach the machine learning model, and we already know the output. Test set can be defined as a subset of the dataset to check the machine learning model, and by using the test set, the model predicts the output.

3.2 Development of the machine learning models

Figure 2 shows schematics of six models that are trained using six machine learning algorithms which are as follows:

Random Forest Classifier [11]: It's a classifier that contains several decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. Instead of relying on one decision tree, the random forest takes the prediction from each tree and supports the majority votes of predictions, it predicts the ultimate output.

K Nearest Neighbor [17]: It's one of the simplest Machine Learning algorithms based on the Supervised Learning technique. The K-NN algorithm assumes the similarity between the new case/data and available cases and puts the new case into the category that's most similar to the

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 10, October-2022

available categories. The K-NN algorithm is usually used for Classification problems. K-NN may be a non-parametric algorithm, which suggests it does not make any assumption on underlying data. It's also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.

Gradient Boosting: This algorithm is one of the most powerful algorithms in the field of machine learning. The errors in machine learning algorithms are broadly classified into two categories i.e. Bias Error and Variance Error. As gradient boosting is one of the boosting algorithms it is used to minimize the bias error of the model. It is often used for predicting not only continuous target variables (as a Regressor) but also categorical target variables (as a Classifier). When it's used as a classifier then the cost function is Log loss.

Logistic Regression [17]: It's one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. it's used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression can work on categorical variables. The results are often either Yes or No, 0 or 1, True or False, etc. but rather than giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1. The values from 0.5 to 1 are often considered as 1 and 0 below 0.5.

Naive Bayes [17]: It's one of the fast and easy ML algorithms to predict a class of datasets. Naive Bayes may be a generative model. It's a probabilistic classifier, which suggests it predicts based on the probability of an object. We've used Gaussian Naive Bayes in our work. (Gaussian) Naive Bayes assumes that every class follows a Gaussian distribution.[16]

Support Vector Machine [17]: It's a very popular Supervised Learning algorithm; it is employed for Classification as well as Regression problems. The goal of the SVM algorithm is to make the best line or decision boundary that can divide n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is named a hyperplane. SVM chooses the acute points/vectors that help in creating the hyperplane. These extreme cases are called support vectors.

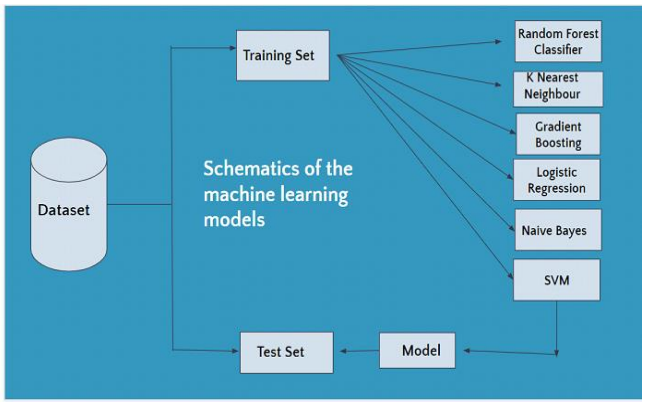

Figure 2. Schematics of the machine learning models

Dataset was splitted in 70% and 30% for training and testing purposes respectively. After training the models they were also tested on the surveyed data that is on Indian population. Predictions were made using the model giving highest accuracy on Indian data.

3.3 Evaluation of the machine learning models

The models are evaluated using accuracy score, confusion matrix and classification report back to assess the reliability of the proposed machine learning models. The results of these measures are compared and the model with the best results in all the aspects has been chosen for the integration purpose in the android application. These metrics are calculated on the idea of the following:

True Positives(TP)- This is often the portion of the dataset in which the patients who were covid positive were correctly identified by the model.

True Negatives(TN)- this is often the portion of the dataset in which the patients who were covid negative were correctly identified as negative by the model.

False Positives(FP)- This is often the portion of the dataset in which the patients who were covid negative were incorrectly identified as positive by the model.

False Negatives(FN)- This is often the portion of the dataset in which the patients who were covid positive were incorrectly identified as negative by the model.

Confusion matrix

A confusion matrix may be a table that is used to define the performance of a classification algorithm. A confusion matrix visualizes and summarizes the performance of a classification algorithm. From our confusion matrix, we can calculate five different metrics measuring the validity of our model.

1. Accuracy (all correct / all) = TP + TN / TP + TN + FP + FN
2. Misclassification (all incorrect / all) = FP + FN / TP + TN + FP + FN
3. Precision (true positives / predicted positives) = TP / TP + FP
4.Sensitivity/ Recall (true positives / all actual positives) = TP / TP + FN
5.Specificity (true negatives / all actual negatives) =TN / TN + FP

A classification report is also a performance evaluation metric in machine learning. It's used to show the precision, recall, F1 Score, and support of the trained classification model.

Accuracy score: This measures the share of correctly identified cases relative to the entire dataset. The ML algorithm performs better if the accuracy is higher.

Precision: This metric measures the exactness, which may be computed as the ratio of true positives to the sum of true and false positives.

Recall: This metric may be a measure of completeness, which may be computed as the ratio of true positives to the sum of true positives and false negatives.

F1 Score: It is often described as the weighted harmonic

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 10, October-2022

mean of precision and recall. The closer the worth of the F1 score is to 1.0, the higher the expected performance of the model is.
Support: It is often described as the weighted harmonic mean of precision and recall. The closer the worth of the F1 score is to 1.0, the higher the expected performance of the model.

3.4 Development of android application

The android app developed is the user interface which has been developed using kotlin with XML as the frontend language for the user interface. The app has been developed on android studio for android 11 (API level 30) with minimum SDK Version 16 and target SDK Version 32. The min SDK version is the earliest release of the Android SDK that our application can run on. Usually, this is often because of a problem with the earlier APIs, lacking functionality, or another behavioral issue. The target SDK version is the version our application was targeted to run on. Ideally, this is often because of some sort of optimal run conditions. This is often mostly to indicate how current our application is for use in the marketplace, etc.

The workflow and therefore the implementation of the application can be depicted with the help of the diagram below in Figure 3.

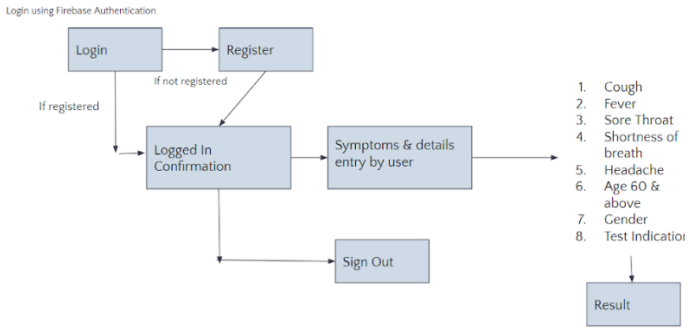

Figure 3. Workflow diagram of the android app

Firebase Authentication: The login and signup system in the app is formed using the Firebase authentication service and the Cloud Firestore of Firebase.
API: The API of the machine learning model is formed using flask and other libraries like gunicorn, sklearn and pickle.
API Deployment: The API made is deployed on Heroku.
Model & Android Integration: Communication between the machine learning model and the android app is established using Rest API and volley library to send and receive post requests through HTTPS.
Fragments: The symptoms and therefore the results have been shown on the same activity with different screens and options every time using the functionality of fragments.
Views: Different text views and button views are used to output and input texts and options to and from the user respectively.
The user requires an android phone and internet to put in the application and use its services. Once the user installs the app, it starts with a splash screen of CoviExpert followed by a login page posing for user credentials (Email Id and Password) after which he/she is taken to a category page. If

the user isn't registered already, the “Wrong Details" message is displayed and therefore the user is required to register using the signup facility provided which will lead him/her to the Login confirmation page to inform him/her about getting authenticated. Google Firebase Authentication and Cloud Firestore services are used to provide this functionality. Here, the user can proceed by clicking thereon which will further take the user to a new screen giving him the option either to sign out or proceed to enter the symptoms and basic details. If the user clicks on the sign out, his session will expire and he is going to be logged out from the firebase authentication account and the device will be free for the new user to log in. If the user opts for entering the symptoms and details, the symptoms are going to be displayed one by one on new screens for the user to select from the options. The user can attend to the next and previous symptoms by using the next and previous buttons respectively. After filling all the small print, the user can confirm by clicking on submit button and the next screen will be displayed which will provide the prediction result and the details. Here the user can click on the small print to know about the technical details and the algorithms used. this may tell the user about the accuracy of the prediction made by the app.

3.5 Integration of app with model

To establish a connection between the android app and the machine learning model is chosen, the Rest API of the model is made using Flask with gunicorn, sklearn, and pickle among others as some of the main libraries. Gunicorn allows the API to handle multiple users at the same time. Firstly, the pkl file of the model is made using the pickle library. Pycharm is used to create API and the pkl file is imported into the project. The post method is used to send and receive HTTP requests. Here, the postman is used to test the API. Once the API is made, it is deployed on Heroku[5] using git and Heroku CLI. After deployment of the API, it is ready to be called in the android app and predicts the results. Volley library is used to send and receive HTTP requests from API.

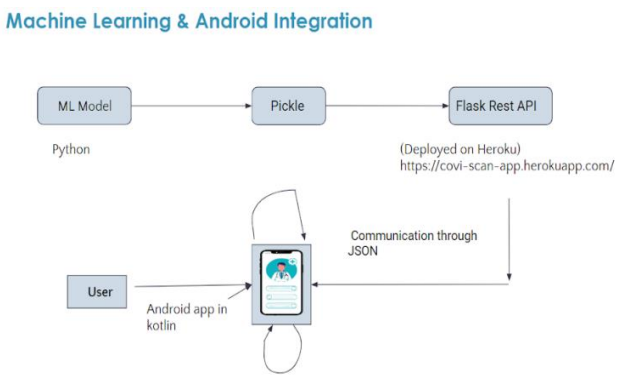

Figure 4. Machine Learning Model and Android App Integration

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 10, October-2022

Figure 4 depicts the whole working of the API and how it can be used to establish communication between the machine learning model in python and the android app in kotlin.

4 RESULTS

Using different evaluation metrics, different outcomes of the models have been observed.

Correlation matrix can be used to observe relationships and correlations between the features used in the model. A correlation matrix is a table which displays the correlation coefficients for different variables. The matrix depicts the correlation between all the possible pairs of values in a table. It is a powerful tool to summarize a large dataset and to identify and visualize patterns in the given data. The correlation matrix for the Israeli dataset used can be seen in Figure 5.

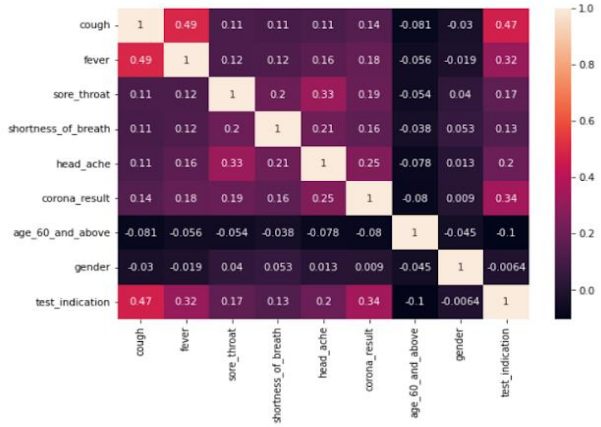

Figure 5. Correlation Matrix for the Israeli Dataset

Each of the features is completely related to itself, therefore the correlation coefficient is 1 in diagonals. -1 indicates a perfectly negative linear correlation between two variables whereas 0 indicates no linear correlation between two variables. The further away the correlation coefficient is from zero, the stronger the relationship between the two variables. Positive value indicates that the features are directly related and negative value indicates inverse relation.

Logistic Regression

The classification report of logistic regression on Israeli test dataset indicates support of 78045 tuples to be negative, 4448 tuples to be positive and 1162 tuples to be of other category with precision, F1-score and recall values of 0.95, 0.97, 0.99 for the negative; 0.78, 0.55, 0.43 for the positive and 0.00, 0.00, 0.00 for the others respectively. The accuracy of the model has been found to be 94.98%. The confusion matrix is given in Figure 6.

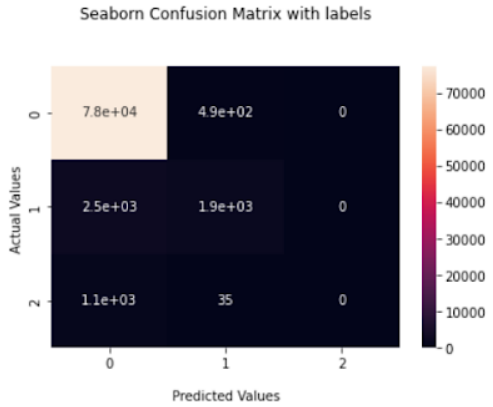

Figure 6. Confusion Matrix as per results by Logistic Regression on Israeli Dataset

The classification report of logistic regression on survey dataset indicates support of 105 tuples to be negative and 44 tuples to be positive with precision, F1-score and recall values of 0.85, 0.82, 0.80 for the negative and 0.58, 0.62, 0.66 for the positive respectively. The accuracy of the model has been found to be 75.83%. The confusion matrix is given in Figure 7.

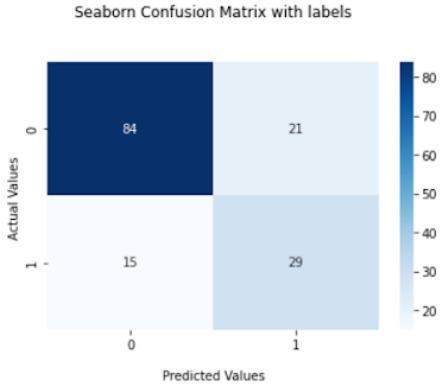

Figure 7. Confusion Matrix as per results by Logistic Regression on Survey Dataset

K-Nearest Neighbor(KNN)

The classification report of KNN on Israeli test dataset calculates precision, F1-score and recall values of 0.95, 0.97, 0.99 for the negative; 0.80, 0.54, 0.41 for the positive and 0.00, 0.00, 0.00 for the others respectively. The accuracy of the model has been found to be 94.96%. The confusion matrix is given in Figure 8.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 10, October-2022

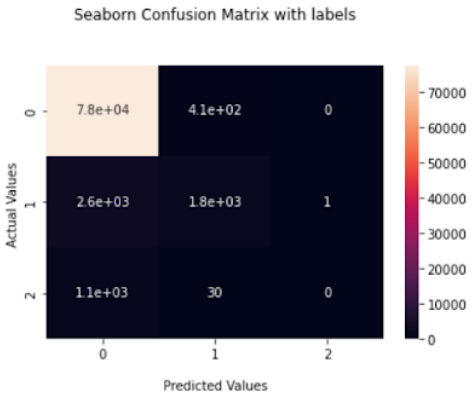

Figure 8. Confusion Matrix as per results by KNN on Israeli Dataset

The classification report of KNN on survey dataset calculates precision, F1-score and recall values of 0.86, 0.77, 0.70 for the negative and 0.51, 0.60, 0.73 for the positive respectively. The accuracy of the model has been found to be 71.14%. The confusion matrix is given in Figure 9.

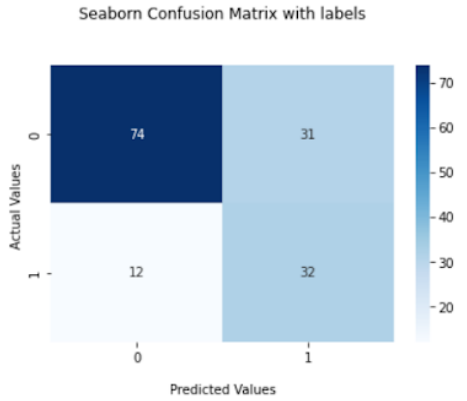

Figure 9. Confusion Matrix as per results by KNN on Survey Dataset

Gradient Boosting

The classification report of gradient boosting on Israeli test dataset calculates precision, F1-score and recall values of 0.96, 0.98, 0.99 for the negative; 0.79, 0.67, 0.58 for the positive and 0.00, 0.00, 0.00 for the others respectively. The accuracy of the model has been found to be 95.59%. The confusion matrix is given in Figure 10.

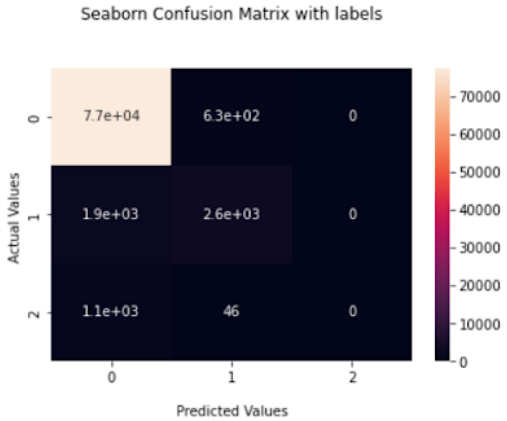

Figure 10. Confusion Matrix as per results by Gradient Boosting on Israeli Dataset

The classification report of gradient boosting on survey dataset calculates precision, F1-score and recall values of 0.83, 0.85, 0.88 for the negative and 0.66, 0.61, 0.57 for the positive respectively. The accuracy of the model has been found to be 72.48%. The confusion matrix is given in Figure 11.

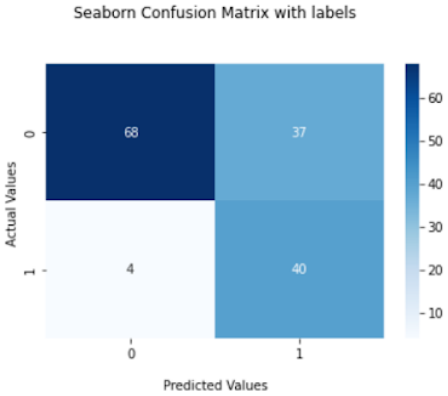

Figure 11. Confusion Matrix as per results by Gradient Boosting on Survey Dataset

Random Forest Classifier(RFC)

The classification report of RFC on Israeli test dataset calculates precision, F1-score and recall values of 0.95, 0.97, 1.00 for the negative; 0.89, 0.39, 0.25 for the positive and 0.00, 0.00, 0.00 for the others respectively. The accuracy of the model has been found to be 94.47%. The confusion matrix is given in Figure 12.

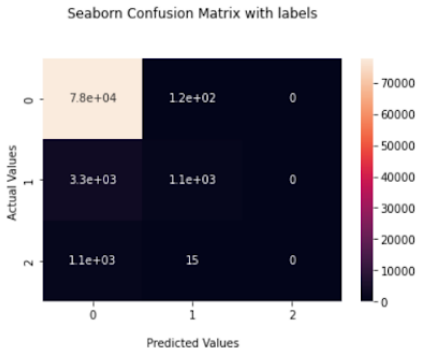

Figure 12. Confusion Matrix as per results by RFC on Israeli Dataset

The classification report of RFC on survey dataset calculates precision, F1-score and recall values of 0.83, 0.85, 0.88 for the negative and 0.66, 0.61, 0.57 for the positive respectively. The accuracy of the model has been found to be 78.52%. The confusion matrix is given in Figure 13.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 10, October-2022

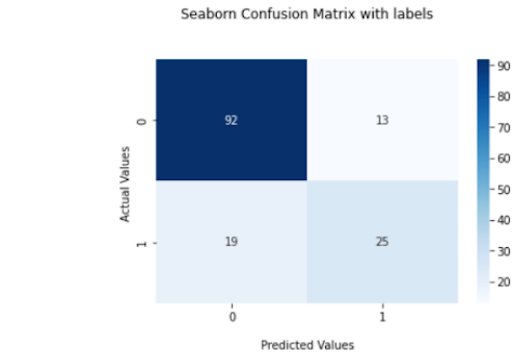

Figure 13. Confusion Matrix as per results by RFC on Survey Dataset

Gaussian Naive Bayes(GNB)

The classification report of GNB on Israeli test dataset calculates precision, F1-score and recall values of 0.97, 0.97, 0.97 for the negative; 0.57, 0.61, 0.65 for the positive and 0.00, 0.00, 0.00 for the others respectively. The accuracy of the model has been found to be 94.19%. The confusion matrix is given in Figure 14.

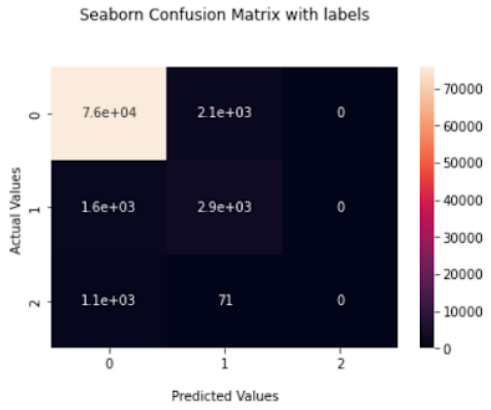

Figure 14. Confusion Matrix as per results by GNB on Israeli Dataset

The classification report of GNB on survey dataset calculates precision, F1-score and recall values of 0.94, 0.77, 0.65 for the negative and 0.52, 0.66, 0.91 for the positive respectively. The accuracy of the model has been found to be 72.48%. The confusion matrix is given in Figure 15.

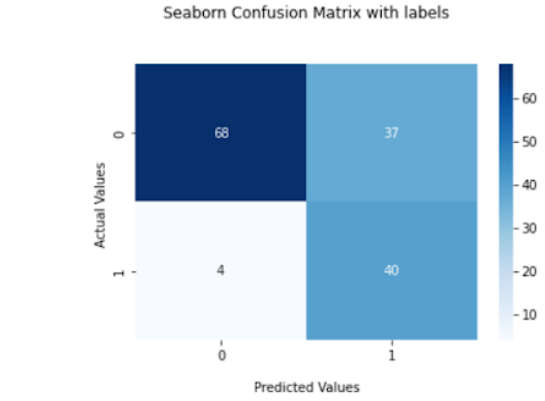

Figure 15. Confusion Matrix as per results by GNB on Survey Dataset

Support Vector Machine(SVM)

The classification report of SVM on Israeli test dataset calculates precision, F1-score and recall values of 0.94, 0.97, 0.97 for the negative; 0.57, 0.61, 0.65 for the positive and 0.00, 0.00, 0.00 for the others respectively. The accuracy of the model has been found to be 94.41%. The confusion matrix is given in Figure 16.

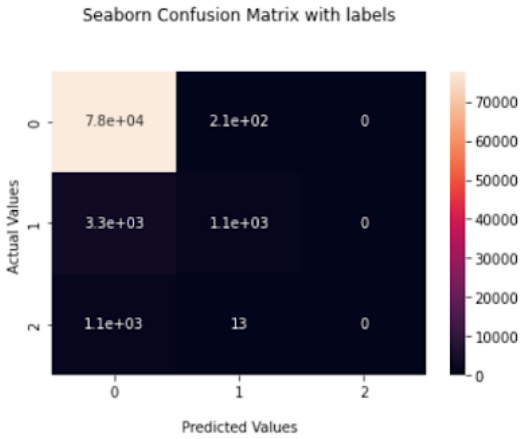

Figure 16. Confusion Matrix as per results by SVM on Israeli Dataset

The classification report of SVM on survey dataset calculates precision, F1-score and recall values of 0.91, 0.79, 0.70 for the negative and 0.54, 0.65, 0.84 for the positive respectively. The accuracy of the model has been found to be 73.82%. The confusion matrix is given in Figure 17.

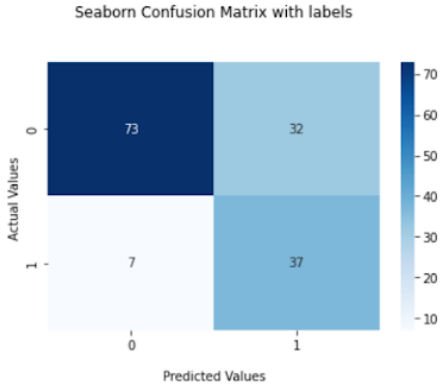

Figure 17. Confusion Matrix as per results by SVM on Survey Dataset

5 DISCUSSION

Table1 gives the summary of the results of all the models.

Model	Accuracy on Israel Dataset(%)	Accuracy on Indian Dataset(%)	F1-score (Israel Dataset) (Weighted Average)	Precision (Israel Dataset) (Weighted Average)	Recall (Israel Dataset) (Weighted Average)
Logistic Regression	94.98	75.83	0.94	0.93	0.95
K-Nearest Neighbor	94.96	71.14	0.94	0.93	0.95
Gradient Boosting	95.59	72.48	0.95	0.94	0.96
Random Forest Classifier	94.46	78.52	0.93	0.93	0.94
Support Vector Machine	94.19	72.48	0.94	0.93	0.94
Naive Bayes	94.41	73.82	0.94	0.93	0.94

Table 1. Summary of the results of machine learning models

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 10, October-2022

Weighted average of F1 score, precision and recall is used to compare the models since the dataset was imbalanced. From the above summary, it can be concluded that the model with highest accuracy on the test dataset of Israel is gradient boosting while the model with highest accuracy on the survey dataset is random forest classification. Logistic Regression also performs quite well on both the test datasets. Accuracy of the models seemed to degrade for the survey dataset which might be because of the regional factors and less data as well. Gradient Boosting doesn't seem to perform well for survey dataset. Random forest is less prone to overfitting. Therefore, the model chosen to predict results on the android application was random forest classifier since it gives better accuracy on the individuals tested here and also, random forest seemed to be the best alternative for deployment as an API as it causes less complications while integration as compared to gradient boosting and hence predicts results faster.

Figure 18 shows the procedure of selection of the best model.

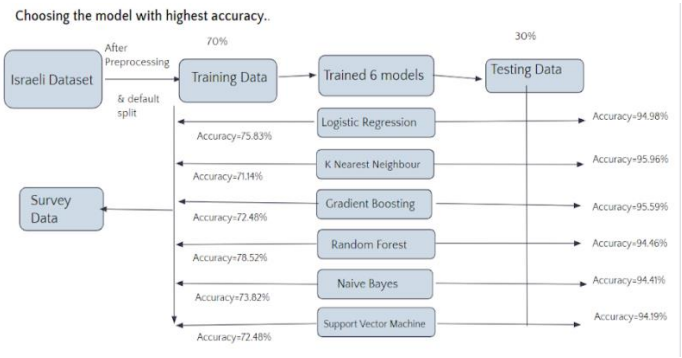

Figure 18. Procedure for choosing the best model

6 CONCLUSION AND FUTURE SCOPE

This system is capable of predicting the covid -19 with high accuracy it can contribute largely in developing a self-monitoring platform for people by alerting the users whether or not to visit the doctor. Such a system can decrease the rush at OPDs of hospitals and reduce the workload on medical staff. Such a system can be relied on to do the job with the easy to use interface.

As a future scope of this model, this model can be made to make more real time predictions using sensors compatible with the attributes and the IoT system. The accuracy can be made more accurate for all the regions by collecting more data which is validated and is diverse. This can be expanded further to other diseases and healthcare guidelines. Changes and optimizations can be made to enhance the speed and working of the application.

REFERENCES

[1] Ahmad Alimadadi, Sachin Aryal, Ishan Manandhar, Patricia B. Munroe, Bina Joe, and Xi Cheng. (2020). Artificial intelligence and machine learning to fight COVID-19. Physiological genomics, 52(4), pp. 200-202.

[2] Pagadala Suganda Devi. (2017). Research methodology: A handbook for beginners. Notion Press. ISBN 978-1-947752-84-9. Pages 186.

[3] Israeli Dataset. Data.gov.il. (2022). [online] Available at: https://data.gov.il/dataset/covid-19

[4] Survey Dataset. https://github.com/Arushi1918/Covid-19-Prediction-App

[5] API of covid prediction app. Herokuapp.com. (2022). [online] Available at: https://covi-scan-app.herokuapp.com/

[6] Vihakari, M., May 17, 2020. GitHub - MikkoVihtakari/COVID-19-app: A shiny app to predict and study the beginning phase of COVID-19 outbreak. [online] GitHub. Available at: https://github.com/MikkoVihtakari/COVID-19-app.

[7] Beatrice Kennedy, et al. (2022). App-based COVID-19 syndromic surveillance and prediction of hospital admissions in COVID Symptom Study Sweden. Nature communications, **13**(1), pp. 1-12.

[8] Mojjada, R. K., et al. (2020). Machine learning models for covid-19 future forecasting. Materials Today: Proceedings. doi:10.1016/j.matpr.2020.10.962

[9] H, W.K. (2020). COVID-19 Outbreak Prediction using Machine Learning Algorithm. [online] Medium. Available at: https://towardsdatascience.com/covid-19-outbreak-prediction-using-machine-learning-algorithm-ce5641bd55bf

[10] Celestine Iwendi, Ali Kashif Bashir, Atharva Peshkar, R. Sujatha, JyotirMoy Chatterjee, Swetha Pasupuleti, Rishita Mishra, Sofia Pillai and Ohyun Jo. (July 2020). COVID-19 Patient Health Prediction Using Boosted Random Forest Algorithm. Frontiers in public health, 8, 357.

[11] Breiman, L. (2001). Random forests. Machine learning, 45(1), pp. 5-32.

[12] Shobana, M., Vaishnavi, S., Prasad, C.G., Poonkodi, P., Sabitha, R. and Karthik, S. (2022). Relating Design Thinking Framework in Predicting the Spread of COVID in Tamilnadu Using ARIMA. Communications in Computer and Information Science. [online] doi:10.1007/978-3-030-95502-1_1.

[13] Kırbaş, İ., Sözen, A., Tuncer, A.D. and Kazancıoğlu, F.Ş. (2020). Comparative analysis and forecasting of COVID-19 cases in various European countries with ARIMA, NARNN and LSTM approaches. Chaos, Solitons & Fractals, 138, p.110015. doi:10.1016/j.chaos.2020.110015.

[14] Alali, Y., Harrou, F. and Sun, Y. (2022). A proficient approach to forecast COVID-19 spread via optimized dynamic machine learning models. Scientific Reports, [online] 12(1). doi:10.1038/s41598-022-06218-3.

[15] Anon, (n.d.). Data Preprocessing in Machine learning – Shishir Kant Singh. [online] Available at: https://shishirkant.com/data-preprocessing-in-machine-learning-2

[16] Bustamante, C., Garrido, L., Soto, R. (2006). Comparing Fuzzy Naive Bayes and Gaussian Naive Bayes for Decision Making in RoboCup 3D. In: Gelbukh, A., Reyes-Garcia, C.A. (eds) MICAI 2006: Advances in Artificial Intelligence. MICAI 2006. Lecture Notes in Computer Science(), vol 4293. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11925231_23

[17] Soofi, A. A., & Awan, A. (2017). Classification techniques in machine learning: applications and issues. Journal of Basic & Applied Sciences, 13, 459-465.