**Department of Electrical and Computer Engineering**

**North South University**



**Senior Design Project 499A Report**

**COVID-19 Prediction using ML**

**Sanzida Solayman ID # 1631277042**

**Sk. Azmiara Aumi ID # 1722263642**

**Chand Sultana Mery ID # 1631500042**

**Muktadir Mubassir ID # 1813243642**

**Faculty Advisor:**

**Riasat Khan**

**Assistant Professor**

**ECE Department**

**Fall, 2021**

**COVID-19 Prediction using ML**

**Individual Contribution Table**

|  |
| --- |
| **Section** |
| IEEE/LaTEX formatting |
| Turnitin check |
| Grammarly check |
| Abstract |
| Keywords |
| Introduction Motivation |
| Paper Review 1 |
| Paper Review 2 |
| Paper Review 3 |
| Paper Review 4 |
| Proposed System |
| Results and Discussion |
| Conclusions |

COVID-19 Prediction using ML

Sanzida Solayman  
Department of Electrical and Computer Engineering  
North South UniversityDhaka, Bangladesh  
sanzida.solayman16@northsouth.eduSk. Azmiara Aumi  
Department of Electrical and Computer Engineering  
North South UniversityDhaka, Bangladesh  
sk.azmiara@northsouth.edu Chand Sultana Mery  
Department of Electrical and Computer Engineering  
North South UniversityDhaka, Bangladesh  
sultana.mery@northsouth.edu Muktadir Mubassir  
Department of Electrical and Computer Engineering  
North South UniversityDhaka, Bangladesh  
muktadir.mubassir@northsouth.edu

*Abstract*—COVID-19 is currently an extremely dangerous and global concern. COVID-19 is spread from one infected person’s body to another through coughing and sneezing, and it spreads when people breathe in air contaminated with droplets and minute airborne particles. The risk of inhaling these is greatest when people are close together, although they can be inhaled over greater distances, especially indoors.This paper has focused on analyzing COVID-19 shambles as well as detection of COVID-19 using different Machine Learning techniques to build up a web app. The dataset has been preprocessed by dropping null values, feature engineering, and under-sampling. We have trained and evaluated different types of classifiers like Logistic Regression, Random Forest, Decision Tree, KNN model. Among them, the Decision Tree and Random Forest models are performing slightly better than other models on the dataset.

Keywords— COVID-19, Machine learning, Decision Tree, Logistic Regression, KNN, Random Forest, Classification, Webpage.

# Introduction

The coronaviruses are a group of viruses that is incredibly diverse with a wide variety of variants that vary in a considerable number of metrics. Scientists had first discovered a strain of the coronavirus that infects humans in 1965. This strain caused common colds in their host body. More than a decade later, research unearthed a group of viruses found in humans and animals, called by their looks that resemblance a crown, the word ‘corona’ meaning ‘crown’ in Latin. Thus far, scientists contend that up to seven coronaviruses can infect people. In 2002 one such severe acute respiratory syndrome-related virus was originally discovered for the first time in the south of China, which was surprisingly quick to pass on in about 30 more countries. However, the question that arises here is how this virus became the cause of global devastation, in the form of the COVID-19 pandemic. Specialists opine that SARS-CoV-2 originated in bats. In Wuhan’s wet-market, people would pay a visit to purchase fish and fresh meat as the animals were slaughtered in the same place. It is believed this is from where the contamination was passed onto humans. In some wet markets, there is trade of prohibited and endangered species such as cobras, wild boards and racoon dogs. The congested and crowded environment facilitates cross-contamination and the swapping of genes between various animals. This may result in viruses undergoing significant mutation, with the potential to infect humans and propagate rapidly and devastatingly [1].

The coronavirus kept spreading all over the world very rapidly. According to the statistics of 13 January 2022, there are 317,983,061 Coronavirus cases all around the world causing 5,533,431 deaths. The statistics in our country are not good either, there are 1,604,664 cases of the Coronavirus in Bangladesh, with the death number being 28,123. [2] Because the diagnostic technology is not available everywhere, performing a rapid test as an alternative to identifying COVID-19 is currently deemed difficult. Given the limits of COVID-19 testing, additional diagnostic methods are desperately needed. The major goal of developing this web tool is to ensure that users can quickly determine whether or not they are affected by COVID-19 by answering a few questions. People can use forums to ask questions about their symptoms and health concerns, as well as receive advice on possible remedies. This website gives a broad overview of any new respiratory viruses.

In [3], the authors have worked with the data of the suspected COVID-19 patients and their admission rate into different types of beds in the hospital. To train machine learning models, the authors have used data donated by the Hospital Israelita Albert Einstein, Sao Paulo, Brazil, and the data contains the clinical information of the suspected and COVID-19 patients such as CBC (Complete Blood Cell counts) and data related to liver, glucose and renal test of the suspected and COVID-19 positive patients. The authors have trained models on the dataset using six machine learning algorithms, namely Logistic regression, Decision tree, Random Forest. Bagging, Gradient Boost, and ADA boosting. Among the models, the Random Forest classified achieved the best training accuracy of 97.17% and testing accuracy of 94.80%. The authors evaluated the machine learning models based on their training accuracy, test accuracy, ROC curve, and confidence score.

In [4], the authors designed an Android application that has been implemented with some training, sharing some information, measurement risk, showing some symptoms, communication tracking, providing quick. The authors collected a dataset from the local clothing RBS Fashion in Bangladesh. The author uses a fuzzy neural network algorithm that is used in the proposed iWorkSafe app to fuse examination data and provide a fancy metric for determining health workers. The authors created an app that will track their employees who are sick and what kind of measures and remedial. The authors build up their software with the Larval PHP framework and employ a database, on the back end to do real-time contact tracing among industry employees. The author’s done their risk Detection by analyzing the acquired data using different machine learning approaches, fusing the generated knowledge using a fuzzy neural network approach, and estimating the risk score which denotes the health status of an employee.

In [5], the authors created a COVID-19 detection system using machine learning methodologies, a tracking system utilizing geo-fencing technologies, and an alerting system using a mobile application. In the application they used Geo-fencing and methodological Triangulation for Data collection and they used to train the model by Logistic Regression, Support Vector Machine and XG Boost algorithms. They used XG Boost for better accuracy than SVC model which reached to heights of 99%. They used ‘step out’ feature to access user’s location for tracking and they used firebase database for stored the data of users. However, from the studies, it was concluded that the app able to successfully detect the presence of COVID-19 based on the user to identify and track symptoms and geo-fences around the user when the user exists his security zone.

This paper [6] discusses about the use of occlusion in detecting COVID-19. The analysis is done by the application of a content-adaptive progressive occlusion analysis (CAPAO) algorithm. To increase accuracy upto 98.33%, a CNN model, in addition to the SVM classifier is employed. The CAPOA algorithm is very effective and clear in its distinction between the goal and the occlude. The test has been conducted in several steps, namely, Scanning the Region of Interest; Analysis of the spatiotemporal context; and, checking reference target and motion constraints. This has brought about extremely accurate results from various models in the range of 78.33% to 98.33%. Notwithstanding this, studies conclude that the COVID-RENet Architecture, which is supplied to the SVM to execute the binary classification, has more accuracy than the CAPAO algorithm. Furthermore, in case any nearby objects to the target resembles the occlude in the visual aspect, it will result in the failure of the algorithm.

In this paper, we have talked about the datasets and all the models we are using, how our webpage is going to look and analyzed the results. The dataset contains individual results of various symptoms, basic patient information such as the date of the test, gender, age, and COVID-19 test results for almost 2 million individuals. We have used four machine learning models including Logistic Regression, Decision Tree, Random Forest, and KNN or K Nearest Neighbors.

The proposed system was discussed in Section (ii) with suitable tables, figures, and flowcharts. Section (iii) contains the most important study findings with all the results and analysis. Finally, Section (iv) concludes the article by providing what we have done in this paper with all the future improvements we are planning to execute to this word.

# Proposed System

## Data

The dataset used in this project is from the public GitHub Repository[7] and was originally sourced from the Israeli Ministry of Health website. The dataset contains individual results of different symptoms, basic information of the patients, and the COVID-19 test results of 27,42,596 patients. There are three types of information available in the dataset. It contains basic information about the patient like the date of the test, Sex(male/female), and if the age of the patient is over 60. Then there are indicators that denote symptoms of the patients. There is information of five symptoms in the dataset, and they are – cough, fever, shortness of breath, sore throat, and head ache. And it also contains the COVID-19 test result and if the patient had any known contact with a COVID-19 patient.

## Data Preprocessing

Before using the dataset for machine learning model training, initial exploration and data preprocessing was done. The dataset had a lot of missing values of the patient’s personal information of age and gender. Setting an average value for the age or gender would have an impact on the total dataset as it might become biased because of the missing values replaced with the average value. As the dataset was huge, the rows with missing values for age and gender were dropped. After dropping the rows with missing values, the size of the dataset was 21,86,227. The age information did not contain the exact age of patients and rather had just indicated if the patients were above 60. So, we dropped the age column. Also, the test date was not usable as a feature to train our dataset as it had no connection with COVID-19 results, so we decided to drop that column too. There were some COVID-19 results that were not confirmed as positive or negative. The rows with not confirmed COVID-19 results were dropped. The final size of the dataset was 21,51,898 where 19,43,172 were COVID-19 negative data and 2,08,726 were COVID-19 positive data. The features we decided to use from the dataset are cough, fever, sore throat, shortness of breath, head ache, gender and test indication. All the features were converted from categorical variables to numerical variables as some of the machine learning models cannot train on label data directly and require to have all the features in numerical variables. Fig. 1 shows the COVID-19 results in relation to the features in the dataset.

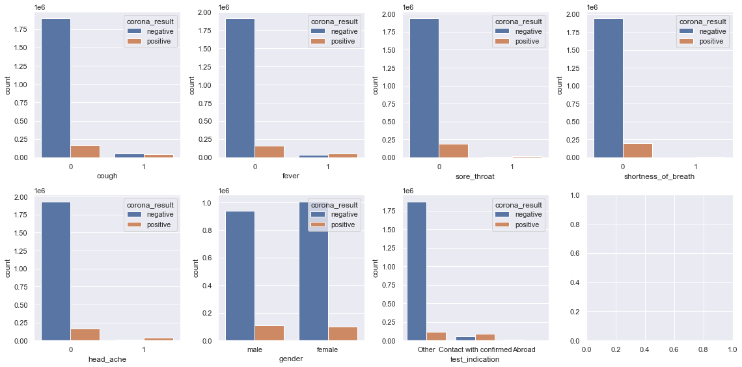


Fig. 1. COVID-19 test results in terms of the features in the dataset.

As Fig. 1 shows, the dataset is highly imbalanced as there is a higher number of negative cases and far a smaller number of positive cases. For this imbalance in the dataset, we used under sampling on the majority class of the dataset and brought the dataset balance to a 100:80 ratio of negative and positive cases. Fig. 2 shows the number of positive and negative cases in the final dataset before and after under-sampling.

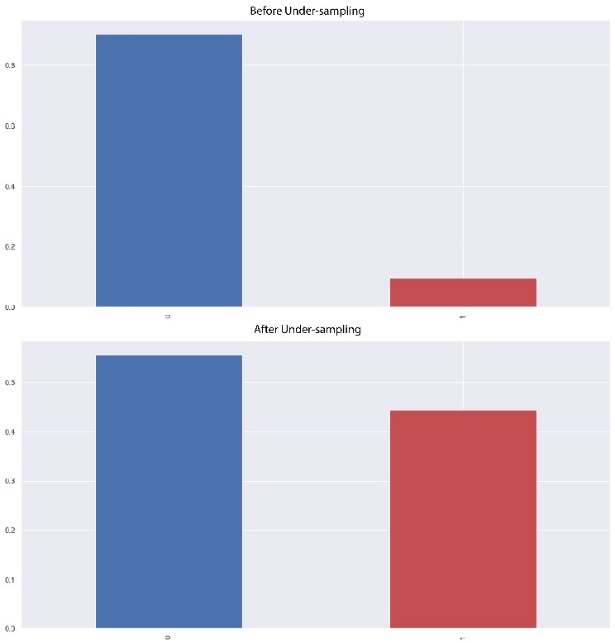


Fig. 2. Positive and Negative case count in the dataset.

## Machine Learning Models

The following four machine learning models have been used so far in this project-

### Logistic Regression: The first machine model used in the project is the popular Logistic Regression model. It is a supervised machine learning model and can be used for classification of two possible classes in a dataset. If we express the Logistic Regression in an equation, it would be-

(1)

### Decision Tree: Decision Tree is one of the popular supervised machine learning models and can be used for regression problem and classification problem, but it is mostly used for classification problems. The Decision Tree classification model is like a tree-structured classifier that has vertex representing the features, points representing the decision rules, and leafs representing the outcome. Fig. 3 is a visual representation of the Decision Tree model’s structure.



Fig. 3. Visual Representation of Decision Tree [9].

### Random Forest: The Random Forest model combines and uses various classifiers to solve regression and classification problems. It utilizes ensemble machine learning techniques. The Random Forest consists of many Decision Trees. It gives the best possible output based on the predictions from all the Decision Trees. Fig. 4 shows the structure of a Random Forest model.

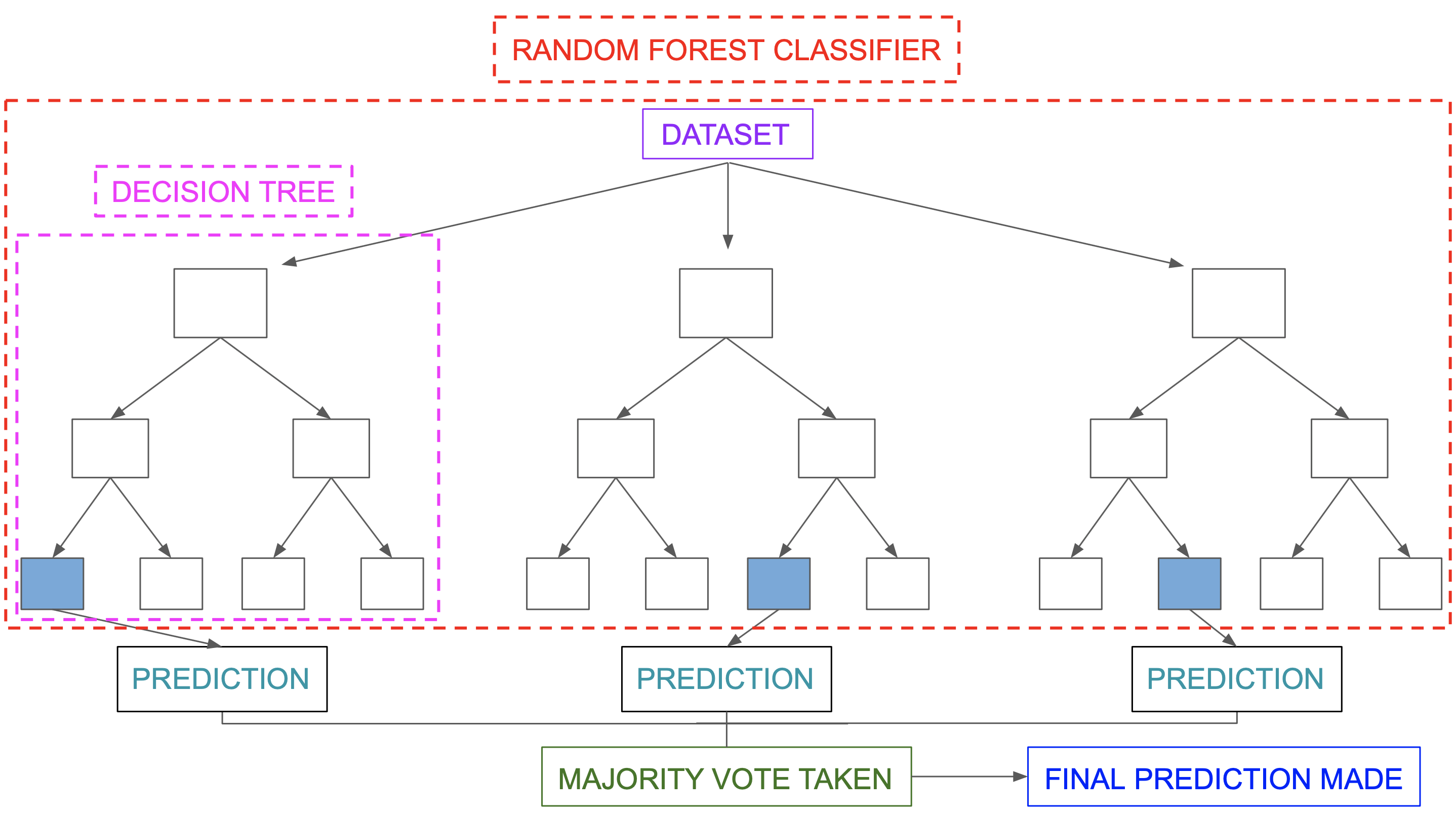


Fig. 4. Random Forest Classification structure [10].

### KNN or K Nearest Neighbors: The KNN is a supervised machine learning model, and it can be used for regression and classification problems. It creates clusters of similar data by calculating their distance and predicts the outcome of new data based on that. Fig. 5 shows how similar types of data exists in close proximity and clustering similar data KNN can predicts the outcome.



Fig. 5. Similar data points typically exist close to each other [11].

After processing the dataset and feature selection, several machine learning models were trained using the dataset. And comparing the performance of the models, the best model is used to predict COVID-19. Fig. 6 gives the visual representation of the flow of training and choosing models in this project.

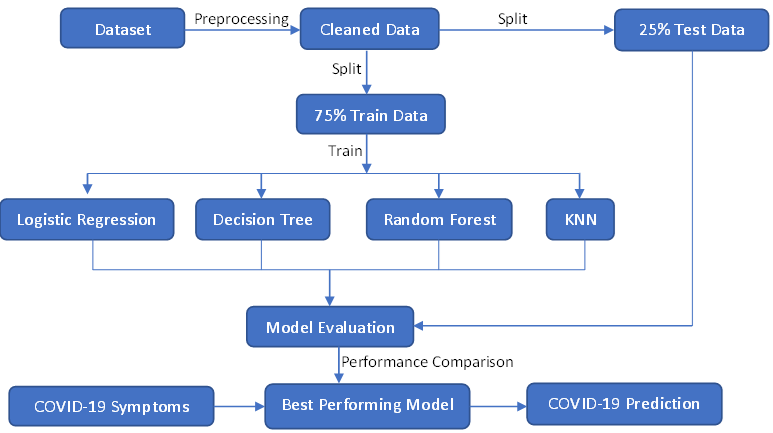


Fig. 6. Flow Chart of training models for this project.

## Webpage

The webpage objective is to give an online COVID-19 detection and remedy referral tool based on the user’s symptoms. People can utilize forums to readily answer questions about their symptoms and health concerns, as well as receive treatment recommendations. There will be a contact page where users can contact us with any questions and queries, which will be directed to the administrators. If a patient tests positive for COVID-19, they can receive suitable treatment options. Our main goal is to raise awareness of people’s COVID-19 situations, which, in the long run, may help prevent the disease from spreading.

To create this web application, we used PHP as the main coding language. PHP’s advantages include improved performance, scalability, built-in security, and simplicity. MySQL was chosen as the backend database because it is one of the most frequently used open-source databases, with quick data access, simplicity, and ease of installation. We used HTML, CSS, and JavaScript for the front wed. JavaScript is a text-based programming language used in both client-side and server-side that allowed us to make web pages interactive. The core framework was designed in HTML, while the style was done in CSS.



Fig. 7. Frontpage of the website.

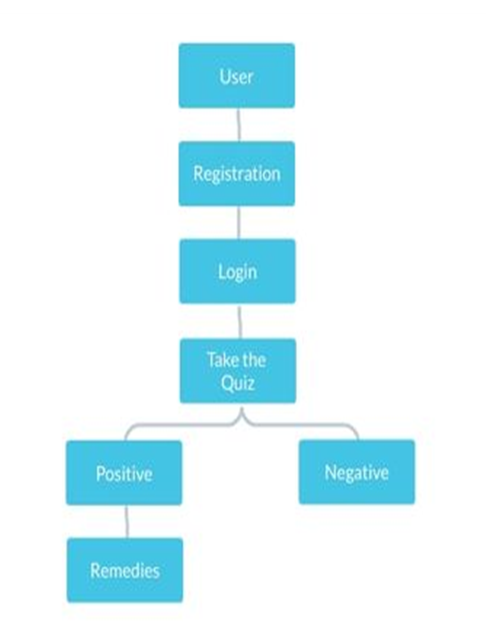


Fig. 8. User flow chart of this web page.

# Results and Discussion

We have split the data into 75% training dataset and 25% test dataset. After training the models on the test dataset, we have evaluated the models on the test dataset based on several performance metrics.

## Logistic Regression Model

On the test dataset, the Logistic Regression model achieved 80.91% accuracy and 83% precision. The model performed better in terms of predicting the positive cases. The f-1 score of the model was 0.80, and recall was 0.81. Fig. 9 shows the confusion matrix for the Logistic Regression model.

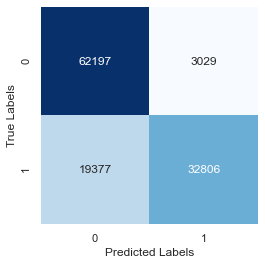


Fig. 9. Confusion Matrix for Logistic Regression Model.

## Decision Tree Model

To find the optimum depth for the Decision Tree model, before training models, we tested the accuracy of different Decision Tree models with the different depths ranging from 1 to 9. The test results showed that depth 7 was optimum for the Decision Tree model with this dataset. Fig. 10 represents the accuracy of the Decision Tree models with different values for depth.

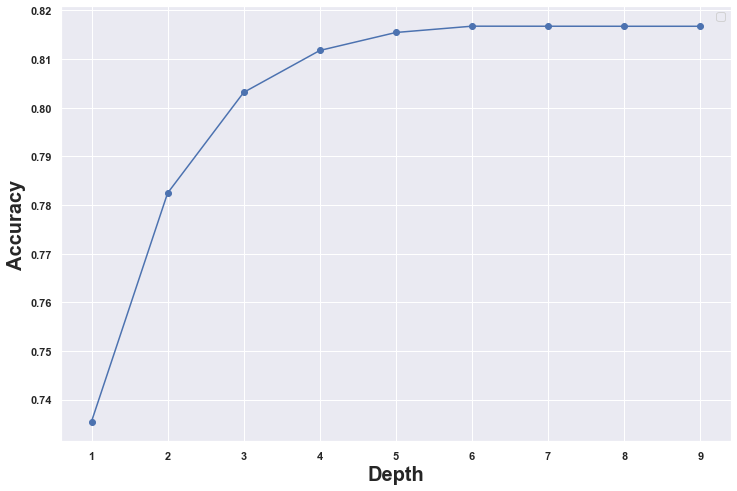


Fig. 10. Accuracy for different depth values of the Decision Tree Model.

The model performed slightly better than the Logistic Regression Model on the test dataset. It achieved 81.67% accuracy and 83% precision on the test dataset. The f-1 score of the model was 0.81, and recall was 0.82.

## Random Forest Model

On the test dataset, the Random Forest model performed similarly to Decision Tree model and achieved 80.91% accuracy with 83% precision. The model’s performance was almost similar to the Decision Tree model. The f-1 score of the model was 0.81, and recall was 0.82. Fig. 11 shows the confusion matrix for the Random Forest model.

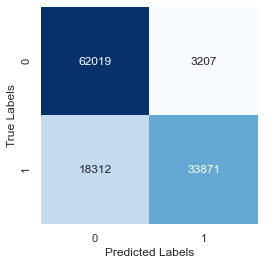


Fig. 11. Confusion Matrix for Random Forest Model.

## K Nearest Neighbors or KNN Model

The KNN model took a longer time than the other models to predict on the test dataset as it was computationally heavy. But the model’s performance was similar to the previous models. It achieved 81.66% accuracy on the test dataset and 83% precision. The f-1 score of the model was 0.81, and recall was 0.82. Fig. 12 shows the confusion matrix for the KNN model.

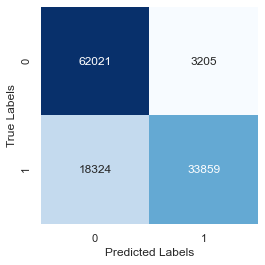


Fig. 12. Confusion Matrix for KNN Model.

## Performance Comparison

To find out the better-performing model, we compared the models’ performances together on several metrics. To get a better understanding of the models’ performances, we plotted the Receiver Operating Characteristic curve or ROC curve and Detection Error tradeoff graph or DET graph of all the models. On the ROC curves, the curves closer to the top left corners indicates better performance. And as we can see from Fig. 13, the ROC curve shows better performance for the Decision Tree and Random Forest models. The DET graph shows the detection error tradeoff of the modes, and it also shows that the Decision Tree and Random Forest performs better than the other models.

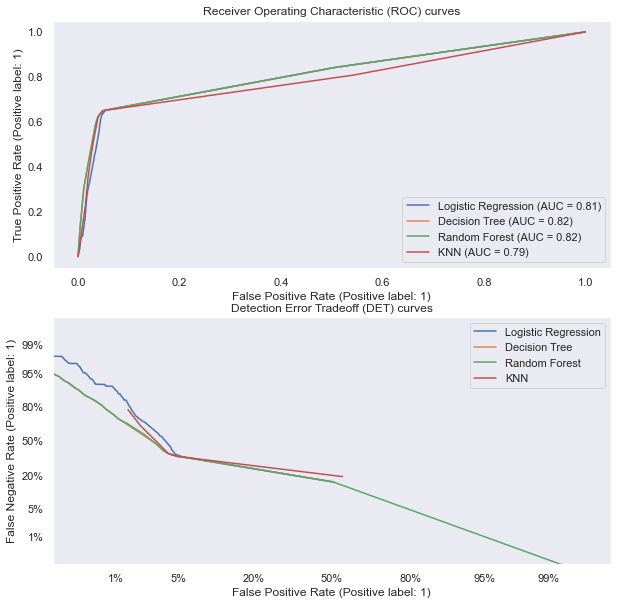


Fig. 13. ROC curve and DET graph of the machine learning models.

From the other performance metrics such as the accuracy, precision, f-1 scores, and recall, we saw that the models’ performances were similar. All of the models performed better in predicting the positive COVID-19 cases than the negative cases. Fig. 14 gives a visual representation of the models’ performances on different metrics.

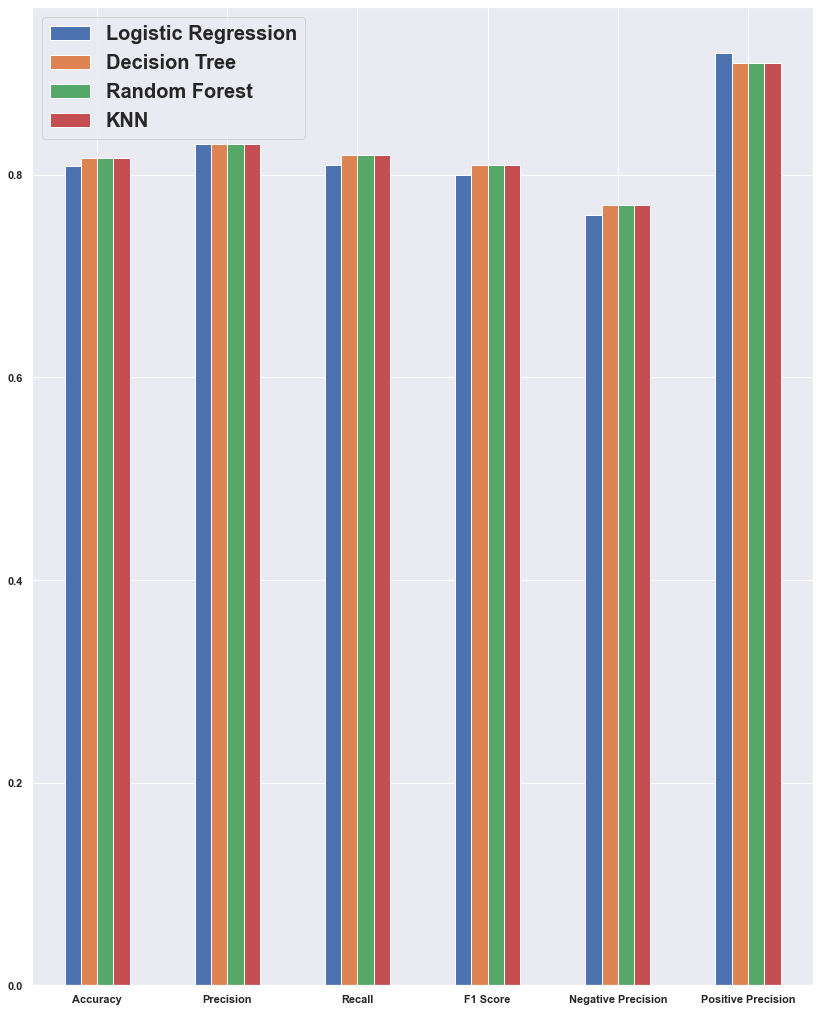


Fig. 14. Comparison of the models’ performances on different metrics.

To evaluate the models’ performances further, we also calculated the Mean Squared Error, Mean Absolute Error, and the r2 score for all the models. And from all the evaluation of the performances of the modes, we can see that the Decision Tree and Random Forest models are performing slightly better than the other models on the dataset so far. Table 1 contains all the scores for the machine learning models.

1. Performance scores of machine learning models

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Performance Metrics** | **Logistic Regression** | **Decision Tree** | **Random Forest** | **KNN** |
| Accuracy | 80.91% | 81.67% | 81.67% | 81.66% |
| Precision | 83% | 83% | 83% | 83% |
| Recall | 0.81 | 0.82 | 0.82 | 0.82 |
| f-1 Score | 0.80 | 0.81 | 0.81 | 0.81 |
| Mean Squared Error | 0.19 | 0.18 | 0.18 | 0.18 |
| Mean Absolute Error | 0.19 | 0.18 | 0.18 | 0.18 |
| r2 score | 0.22 | 0.25 | 0.25 | 0.26 |
| Negative Precision | 76% | 77% | 77% | 77% |
| Positive Precision | 92% | 91% | 91% | 91% |

# Conclusions

The main goal of this project is to predict if the user is infected with the coronavirus and create awareness of people’s COVID-19 circumstances, which may assist in preventing the disease from spreading. In this article, we discussed the datasets and all of the models we will be utilizing, as well as the appearance of our website and how we will analyze the findings through the four machine learning models we have used. The dataset contains records of about 2 million people with their individual symptoms, basic information that includes the date of the test, test result, gender, age, and all the necessary data that is needed to implement the project.

For future work, our main aim is to train the ANN model alongside other models such as the SVM model. On the web front, we will create a database connection with the dataset and use the XAMPP software to host our local host servers such as Apache and MySQL. We will use these database systems to store our data when the registered users take the COVID-19 symptoms quiz or contact us for any queries. Once logged in, the admin will be presented with a list in the MySQL database and can use the data by calling it from the database when needed for result analysis.

##### References

1. Coronavirus History: Origin and Evolution In-text: (COVID-19) et al., 2021).
2. COVID LIVE UPDATE: 317,983,061 CASES AND 5,533,431 DEATHS FROM THE CORONAVIRUS - WORLDOMETER In-text: (COVID Live Update: 317,983,061 Cases and 5,533,431 Deaths from the Coronavirus - Worldometer, 2022).
3. N. Darapaneni, A. Singh, A Paduri, A. Ranjith, A. Kumar, D. Dixit, S. Khan, “A Machine Learning Approach to Predicting Covid-19 Cases Amongst Suspected Cases and Their Category of Admission,” 2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS), 2020, Rupnagar, India, December 26-28, 2020. pp. 375-380, doi: 10.1109/ICIIS51140.2020.9342658
4. M. S. Kaiser et al., “iWorksafe: Towards Healthy Workplaces During COVID-19 With an Intelligent Phealth App for Industrial Settings,” IEEE Access, vol. 9, pp. 13814-13828, 2021.
5. D. Koshti, S. Kamoji, K. Cheruthuruthy, S. P. Shahi and M. Mishra, “A Detection, Tracking and Alerting System for Covid-19 using Geo-Fencing and Machine Learning,” 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), 2021, pp. 1499-1506
6. B. Udawat, A. Santani and S. Agrawal, “Occlusion Detection for COVID-19 Identification: A Review,” 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), 2021, pp. 298-301
7. Nshomron, “NSHOMRON/covidpred: Machine Learning-based prediction of COVID-19 diagnosis based on symptoms,” GitHub. [Online]. Available: https://github.com/nshomron/covidpred. [Accessed: 13-Jan-2022].
8. “Logistic regression in machine learning - javatpoint,” www.javatpoint.com. [Online]. Available: https://www.javatpoint.com/logistic-regression-in-machine-learning. [Accessed: 13-Jan-2022].
9. “Machine learning decision tree classification algorithm - javatpoint,” www.javatpoint.com. [Online]. Available: https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm. [Accessed: 13-Jan-2022].
10. “Introduction to random forest in machine learning,” Section. [Online]. Available: https://www.section.io/engineering-education/introduction-to-random-forest-in-machine-learning/. [Accessed: 13-Jan-2022].
11. O. Harrison, “Machine learning basics with the K-nearest neighbors algorithm,” Medium, 14-Jul-2019. [Online]. Available: https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761. [Accessed: 13-Jan-2022].