Effectivity comparison of machine learning algorithms to detect covid-19

A THESIS

Submitted by

Mehatub Uddin

ID: 2015-3-60-050

Md. Roky Ahmed

ID: 2017-1-60-150

Mahfuzur Rahman

2017-3-60-056

Mir Mumit-Ul-Islam

2016-2-60-108

Department of Computer Science and Engineering East

West University

Dhaka-1212, Bangladesh

February, 2022

Certificate

The research work embodied in the present thesis entitled **"Precision comparison of machine learning algorithms to detect covid-19"** has been carried out in the Department of Computer Science & Engineering, East West University, Bangladesh. The work reported herein is original and does not form part of any other thesis or dissertation based on which a degree or award was conferred on an earlier occasion or to any other scholar.

We understand the University's policy on plagiarism and declare that the thesis and publications are our own work, except where specifically acknowledged and has not been copied from other sources or been previously submitted for award or assessment**.**

MEHATUB UDDIN

AUTHOR

MD.ROKY AHMED

AUTHOR

MAHFUZUR RAHMAN

AUTHOR

MIR MUMIT-UL-ISLAM

AUTHOR

MAHAMUDUL HASAN

SUPERVISOR

SENIOR LECTURER

Dept. of Computer Science & Engineering

East West University

1. Abstract

The covid-19 pandemic has created havoc around the world. Millions of people have been infected and millions are at risk. It is very tough for health organization and government to test each and every citizen for covid test. However, there are many symptoms of this virus. COVID-19 is characterized by a dry cough, sore throat, and fever. Septic shock, pulmonary edema, acute respiratory distress syndrome, and multi-organ failure can occur if symptoms escalate to a severe type of pneumonia with serious consequences such as septic shock, pulmonary edema, acute respiratory distress syndrome, and multi-organ failure. We can use machine learning to detect covid-19 virus. The motive of this paper is to implement four different machine learning algorithms and those are Decision tree, Naïve Bayes, Neural Network and Random Forest to detect covid-19 accurately and recommend the best algorithm, based on the accuracy. Our findings suggest that machine learning can help with COVID-19 inquiry, prediction, and discriminating. Finally, machine learning may be used to analyze and triage COVID-19 instances in health provider programs and plans. With 98 percent testing accuracy of Neural network, Supervised learning outperformed rival Unsupervised learning methods. Recurrent supervised learning may be used in the future to improve accuracy.

Keywords: Covid-19, machine learning algorithm, neural network, naïve Bayes, decision tree, random forest, accuracy.

2. Acknowledgement

We wish to record our deep sense of gratitude and profound thanks to our research supervisor **Mahamudul Hasan**, Senior Lecturer, Department of Computer Science & Engineering, East West University, Dhaka, Bangladesh, for his keen interest, inspiring guidance, constant encouragement with our work during all stages, to bring this thesis into fruition.

We would like to also thank the faculty and non-teaching staff members of the Computer Science & Engineering, East West University, Dhaka, Bangladesh, for their valuable support throughout our research work.

Table of Contents

1. Introduction

Research Background

Technology has advanced rapidly in recent years, and it now plays a vital role in industrialized countries. All aspects of modern life, including education, business, marketing, armies, communications, engineering, and health care, rely on new technological applications. From identifying symptoms to precise diagnosis and digital patient triage, the health care center is a critical sector that requires a strong use of modern technology[1].

Covid-19 has created a global pandemic around the world. It has stopped the whole world. Almost 3.4 million people have died from corona virus or covid-19[2]. Daily new cases increasing. Till date almost 16 crore people got effected from it[3]. So, it is really tough for any government or country to take test for covid-19 for all the population one country has.

Sometimes one country might capable to take test for covid-19 but it is too much time consuming. So, people have to rely on technology for it. As covid virus has some certain symptoms that is sign of corona virus[4].

Problem statement

COVID-19 is now afflicting numerous nations throughout the world. A few nations, like the United States, Germany, Italy, and others, are dealing with the spread of the virus in the community transmission phase, which means that one infected person may infect up to 100 individuals. In these circumstances its very time consuming to test Covid-19 positive or negative. If a person wants to test, he/she might be going outside and it could be risky.

So, given the challenge, the aim is to use machine learning-based model to detect COVID-19 positive or COVID-19 negative. The model uses symptoms as an input parameter, which reveal the disease's early signs.

As a result, in order to address this problem, we developed a model that can detect COVID-19 positive or negative via symptoms.

Objective

The objective of this research is to reduce the testing time. So, that prevention speed will be faster. Also, the accuracy of the test is also an objective so the chances of error is lesser. Here, fast testing is the main priority. A patient or person can do his/her test automatically using device via the symptoms. So, that person don't have to go to hospital for test because it is also increasing the chances of exposing the person to covid-19.

Motivation

As millions of people affected by this virus and medical facility dealing with effected patients. It is getting tougher to get manual test. Which also increases the chances of getting affected. So, the motivation behind this project is to test at home by own,

-To reduce testing time

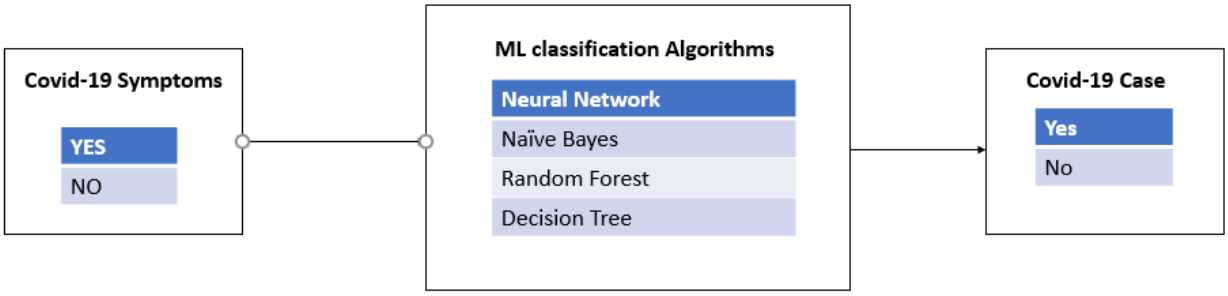
-To determine the affected rate and give medical support quickly.

-To minimize the testing total time, spend on collecting data one by one.

-To give faster and accurate results.

Covid-19 with Machine Learning:

Machine learning is a branch of artificial intelligence that focuses on the development of adaptive computer systems that can improve their performance by incorporating data from previous experiences. Machine learning is the study of generalization, as well as the design and analysis of generalizable algorithms[5]. ML has been effectively used in a variety of applications including finance[6], manufacturing [7], transportation [8], and education[9]. To improve patient health outcomes, AI and ML may be utilized to improve diagnosis, prognosis, monitoring, and therapy delivery[10]. Since the outbreak of COVID-19 began, there has been an increasing interest in employing machine learning to combat the epidemic. In this paper we have tried to detect covid cases via symptoms using some machine learning algorithms. AI and deep learning are essential in finding and classifying COVID-19 cases utilizing computer-aided applications, resulting in great results for identifying COVID-19 instances based on recognized symptoms such as fever, headache, dry cough, and so on. AI and the deep learning model may be used to anticipate the virus's spread based on previous data, which can aid in its management[11].



2. Literature Review

Introduction

The coronavirus disease (COVID-19) caused by the SARS-CoV-2 coronavirus has infected more than 20 million people and has resulted in approximately one million deaths worldwide. To manage this unprecedented pandemic emergency, the early identification of patients using their symptoms is important. Because this early detection of corona virus before the lab test can reduce the infection of this outbreak pandemic. Infectious people is extremely important due to the fact that this disease, unlike others caused by coronaviruses (e.g., SARS, MERS), can coexist in a host organism without causing any symptoms, or it can produce very mild and non-characteristic symptoms in — nevertheless — infectious subjects[12]. To identify SARS-CoV-2 infections, the instrument of choice, or the gold standard, is the molecular test performed using the reverse polymerase chain reaction (PCR) or the reverse transcriptase–PCR (RT-PCR) technique. However, the execution of the test is time-consuming (at no less than 4–5 h under optimal conditions), requires the use of special equipment and reagents, the involvement of specialized and trained personnel for the collection of the samples[12].

To improve our diagnostic capabilities, in order to contain the spread of the pandemic, the data science community has proposed several machine learning (ML) models, solutions based on CT imaging, although accurate, are affected by the characteristics of this modality: CTs are costly, time-consuming, and require specialized equipment; thus, approaches based on this imaging technique cannot reasonably be applied for screening exams.

To overcome the above limitations, and following a successful feasibility study performed on a smaller dataset, we developed different classification models by applying ML techniques.

Related Work

At “Machine learning-based prediction of COVID-19 diagnosis based on symptoms” Zoabi et al. (Author: Yazeed Zoabi , Shira Deri-Rozov and Noam Shomron) Predictions were generated using a gradient-boosting machine model built with decision-tree base-learners. They used the gradientboosting predictor trained with the LightGBM Python package. The validation set was used for early stopping, with auROC as the performance measure. To identify the principal features driving model prediction, SHAP values were calculated in this article. e. By averaging across samples, SHAP values estimate the contribution of each feature to overall model predictions[13]

At “Wearable sensor data and self-reported symptoms for COVID-19 detection” Quer et al. Author(Giorgio Quer, Jennifer M. Radin, Matteo Gadaleta, Katie Baca-Motes , Lauren Ariniello , Edward Ramos, Vik Kheterpal , Eric J. Topol and Steven R. Steinhubl). Receiver operating characteristic curves (ROCs) for the discrimination between COVID-19-positive and COVID-19-negative cases based on the available data: RHR data (a); sleep data (b); activity data (c); all available sensor data (d); symptoms only (e); symptoms with sensor data (f). Models are based on a single decision threshold. Median values and 95% confidence intervals (CIs) for sensitivity (SE), specificity (SP), positive predictive value (PPV) and negative predictive value (NPV) are reported, considering the point on the ROC with the highest average value of sensitivity and specificity. Error bars represent 95% CIs. P values from a one-sided Mann–Whitney U test are reported[14].

At “Development, evaluation, and validation of machine learning models for COVID-19 detection based on routine blood tests” Cabitza et al.(author: Cabitza, Federico) For classification they evaluated five different algorithms: Random Forest (RF), naive Bayes (NB), logistic regression (LR), support vector machine (SVM), and k-nearest neighbors (KNN). We specifically evaluated these algorithms as all have been shown to achieve state-of-the-art performance on tabular data and, at least to some degree interpretable. The hyper-parameters of the different classification algorithms are reported. All hyper-parameters were optimized automatically using a grid search approach.

In regard to model selection, training and evaluation, they performed a two-step procedure to minimize the risk of over-fitting: first, the dataset was split into a training set (80% of the instances) and a hold-out test set (20% of the instances), using a stratified procedure; second, hyper-parameter optimization was performed (on the training set) through 5-fold stratified cross-validation grid search and using AUC as reference measure; third, the models were trained and calibrated on the whole training set; finally, the calibrated models were evaluated on the hold-out test set in terms of accuracy, sensitivity, specificity, AUC, and the Brier score. In all stages of model development, the randomization was controlled in order to ensure repeatability of the experiments[12].

3. Methodology

In this review, we tried to acquire queries from online databases including Kaggle, IEEE Xplore, and Google Scholar. We also attached the reference lists of the related articles to find other co-related studies to include in this review. The search terms included COVID-19, machine learning, diagnosis, mortality, severity, and laboratory. We gave importance on ML-based approaches used for predicting COVID-19 with their symptoms and severity, using only simple clinical and laboratory data that were readily available from a public database.

we separate our main dataset with several proportions such as training dataset and testing dataset. We allocate 80% data of dataset for training and 20% data of main dataset for testing our ML classification model.

ML Classification models

During this project we have taken four popular classification model like

- Decision Tree.
- Naïve Bayes.
- Random Forest.
- Neural Network.

In the Given table below, we will discuss about ML algorithms and their features:

ML algorithm	Basic Idea	Features
DT	Structured tree model	Robust, for categorical data, interruption is easy.
NB	Probabilistic classifier	Stable performance, cannot handle missing data [15].
RF	DT ensemble method	Effectiveness for highly complex problems is good, perfect high dimensional datasets, handles missing data and imbalanced data sets.
NN	Inspired by networks of biological neurons	Accuracy in this model is high, difficult to interpret the model, a large number of parameters required

Decision Tree (DT)

A decision tree is a classifier which expressed as partition of recursive space. Decision tree (DT) is a tree-structured model algorithm for describing the relationships between a class label and attributes[16]. A decision tree has no incoming edge, it consists of nodes which forms rooted tree. Other incoming nodes has one node incoming. Here is a node called internal or test node

which have an outgoing node. Rest of other nodes are called leaves which also known as terminal node or decision node.

In a decision tree every internal node divide the instance space into two or more sub-spaces with respect to a certain discrete function from the input attributes values. In a frequent case each case considers single attributes. So that the space is divided according to the attribute value. When the attribute is numeric the conditions referred to a range.

Every leaf is assigned to one class which represents the appropriate target value. the leaf may hold the probability vector indicating the probability of target attribute that has a certain value. It recursively works by dividing the observations based on the attribute that has more information with the highest gain ratio value calculate as follows:

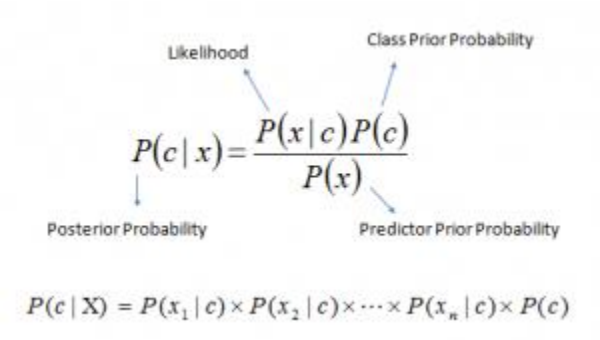
$$\text{gain ratio(A)} = \frac{\text{Gain(A)}}{\text{SplitinfoA(D)}}$$

Here, Splitinfo_A (D) refers possible information provided by partitioning the dataset, D, into V partition and Gain(A) means the amount of information obtained by partitioning the dataset in attribute A [15].

Naïve Byes (NB)

Naïve byes is the simple probabilistic classifier which is based on Byes theorem. In simple words, a Naive Bayes classifier predicts that the presence of a particular feature in a class is unrelated to the presence of any other feature. This model is easy to build and extremely useful for very large data sets. Along with simplicity, it is known to outperform even highly implemented classification methods.

Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c). which is given below:



Here,

- $P(c|x)$ is the posterior probability of *class* (c, *target*) given *predictor* (x, *attributes*).
- $P(c)$ is the prior probability of *class*.
- $P(x|c)$ is the likelihood which is the probability of *predictor* given *class*.
- $P(x)$ is the prior probability of *predictor*.

The word naive refers to the main assumption of conditional independence. Naïve Byes assumes independence between class attributes. This might not be true in real-world applications.

Random Forest

Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset[17]. It is less sensitive to training dataset than other algorithm It predicts output with high accuracy, even for the large dataset it runs efficiently. It works also very good even if there is missing of large portion of data in dataset. Although we have no missing value in our dataset. When we will work

with real time values then it will be more efficient. Random Forest works in two steps first is to create the random forest by combining N decision tree, and second is to make predictions for each tree created in the first phase. The (random forest) algorithm establishes the outcome based on the predictions of the decision trees. It predicts by taking the average or mean of the output from various trees. Increasing the number of trees increases the precision of the outcome[18].

In our work random forest is used thus way mentioned below:

1. We import the scikit learn random forest model library.

```
from sklearn.ensemble import RandomForestClassifier
```

2. We fitted the random forest algorithm to the training set.

```
model = model.fit(x_train, y_train)
```

3. Predict the test set result

```
y_pred = model.predict(x_test)
```

4. initialize variable with ML algorithm

```
rf = RandomForestClassifier(n_estimators=100, max_depth=2,random_state=0)
```

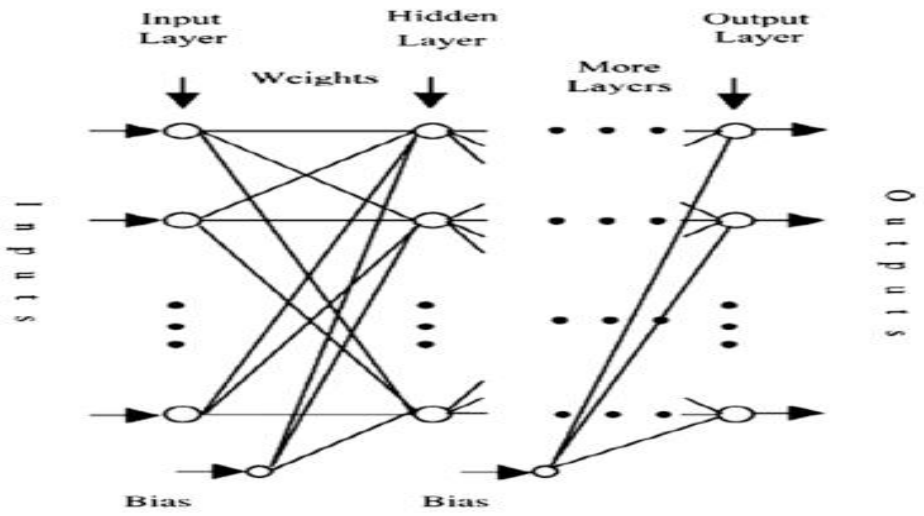
5. predict value and print accuracy score

```
y_pred_rf = modelRun(rf, "Random Forest", x_train, y_train, x_test, y_test)
```

MLP Classifier:

MULTILAYER PERCEPTRONS (MLP) are strong classifiers that may outperform other classifiers, although they are frequently chastised for having too many free parameters. Parameters are often established using either a validation set or cross-validation procedures[19]. It necessitates a series of training sessions followed by a thorough evaluation procedure in order to choose the optimal training[20].ANN Multi-Layer Perceptron Classifier can process with several hidden layer connected between adjacent layer. In MLP, the output values from the previous layer are multiplied by the synaptic weight of the connection and added at the neuron of the next layer. The outcome is the argument of an activation function that represents the neuron output in continuous values. Each unit (neuron) generates an output signal that is proportional to the sum of its input signals. The following is the definition of this function[21]:

$$y_i = f\left(\sum x_i\, w_i\right)$$



4. Empirical analysis

Evaluation Metrics:

There are a number of methods for modeling regularity in the data. For learning result, methods in the same sets of examples changing the parameters of the method in different models [5]. For same set of training data and same problem can create a high number of different models. Which creates the importance of evaluation of quality model with respect to the problem given. That is the reason why evaluation of discovered knowledge is one of the essential components of the intelligent data analysis process. As this work relate with classification problems, we will talk about classification models. To measure the degree the task of evaluating classification model is applied where classification suggest the corresponding model to the actual classification of the case. Applying the method of observing, there are various measures for evaluating the performance of the model. The selection of the most appropriate measures will be done by depending on the characteristics of the problem and its ways of implementation.

Accuracy

The notion of fault in the evaluation of classification models is basic concept. There is an error in classification if the selected case of the application of the classification models is different from the actual class examples. The total number of errors in the observed set used an as indicator of work a classifier when any mistake is equally important. To evaluating the quality of classification model this approach is based on accuracy.

This measure can be defined as the ratio of the number of correctly classified examples according to the total number of classified examples[22].

$$Accuracy = \frac{\text{number of correctly classified examples}}{total\ number\ of\ cases}$$

In accuracy the crucial disadvantages as a measure for evaluation are given:

(1) neglects the differences between the types of errors.

(2) dependent on the distribution of class in the dataset.

As per the confusion matrix we can classify

$$Accuracy = \frac{true\ positive + true\ negative}{true\ positive + false\ positive + false\ negative + true\ negative}$$

Precision

Precision is the measurement of accuracy that counts the percentage of pertinent items from all the items that have been retrieved[23]. For better performance, the measure should be as high as possible. The computed formula is –

$$\text{Precision} = \frac{\text{true positive}}{\text{true positive + false positive}}$$

Here, true positive ratings are those that were predicted as good and the actual rating was also as that. The ratings of positive and negatives were counted within a threshold. The value of both ratings was needed to be measured in precision.

Recall

Recall stands for the complete metric that holds the percentage of the retrieved items excerpt from all the pertinent items. The formula here illustrates as below –

$$\text{Recall} = \frac{true\ positive}{true\ positive + false\ negative}$$

False-negative is the value that is actually rated in the good range but predicted as bad.

F1-Score

The F-Score metric is the comparing harmonic mean of Precision and Recall. As precision and recall evaluations do not provide the significance, f-score invents the balance there and combine both the evaluation for the best performance.

$$\text{F1-score} = \frac{2 \times precision \times Recall}{precision + Recall}$$

5. Performance Assessment

Confusion Matrix

A compatible tool for performance assessment of classification problem is confusion matrix which consists of 4 parts [24]:

1) True -Positive (TP): The number of positive samples predicted to be positive.
2) False- Positive (FP): The number of negative samples predicted to be positive.
3) True -Negative (TN): The number of negative samples predicted to be negative.
4) False- Negative (FN): The number of positive samples predicted to be negative.

In machine learning a confusion matrix, which is also known as an error matrix, is a specific table layout that permits visualization of the performance of a machine learning algorithm, normally a supervised learning one Each row represents the instances in a predicted class of the matrix while each column represents an actual class. The name refers the fact that it makes it easier to see if the system is confusing two classes [25]

A confusion matrix is an N×N matrix which is used for evaluating the performance of a classification model, Here N is the number of target class. The matrix comparison between the target value with the predicted value in the machine learning algorithm.

As we have binary value in our program it will be 2×2 matrix given below:

		Actual Values	
		Positive	Negative
Predicted Values	Negative	TP	FP
	Positive	FN	TN

Fig: Confusion matrix

Roc Curve

Another way of test the performance is Roc curve [26]. This is a two-dimensional graph where X-axis represents the false positive rate and y-axis represent the true positive rate.

In our project item, (0,1) is the perfect classifier which classifies all positive and all negative cases correctly. This is (0, 1), the reason is that the false positive rate is 0 (zero), a positive real rate is 1 (all). For, Point (0, 0) is a classifier that predicts all cases to be negative, while point (1, 1) corresponds to the classifier which indicates that every case is positive. Point (1, 0) is a classifier which is incorrect for all classifications.

In common cases, by increasing the real positive rate the parameter of a classifier can be adjusted at the cost of increasing false positive rates or reducing the false positive rate based on the dropping value of real positive rates. Each setting parameters gives par value for a false positive

rate and positive real rates and the number of such pairs can be used to represent the ROC curves. Nonparametric classifier is presented ROC to one point, which corresponds to the par value of the false positive rate and positive real rate [5].

Some characteristics of Roc curve are:

1. ROC curve is independent of the distribution of the class or the cost of errors [27].
2. Roc curve has all the information contain in the matrix of errors [26]
3. ROC curve gives a graphical tool for testing the ability of the classifier for correcting the positive cases and negative cases which were incorrectly classified.

Prevost and Fawcett in 1997 [28] disagreed that the use of the classification accuracy of the classifier comparison is not sufficient for measuring unless the cost classification and distribution of class is not known, but one of the classifiers should be chosen for each situation. They propose a way of assessing the classifier by using the ROC graph, imprecise costs and distribution of class.

6. Result analysis

Dataset:

Dataset is an essential part of any research paper. This paper worked with the public dataset named ”Covid_Dataset” where covid symptoms is stored[4]. The dataset contains 5434 individual cases where contains 20 symptoms given by the users[4]. This dataset gives an efficient impression to research in accuracy prediction[5].Firstly we have done data pre-processing to cleaning the dataset to avoid data noise.

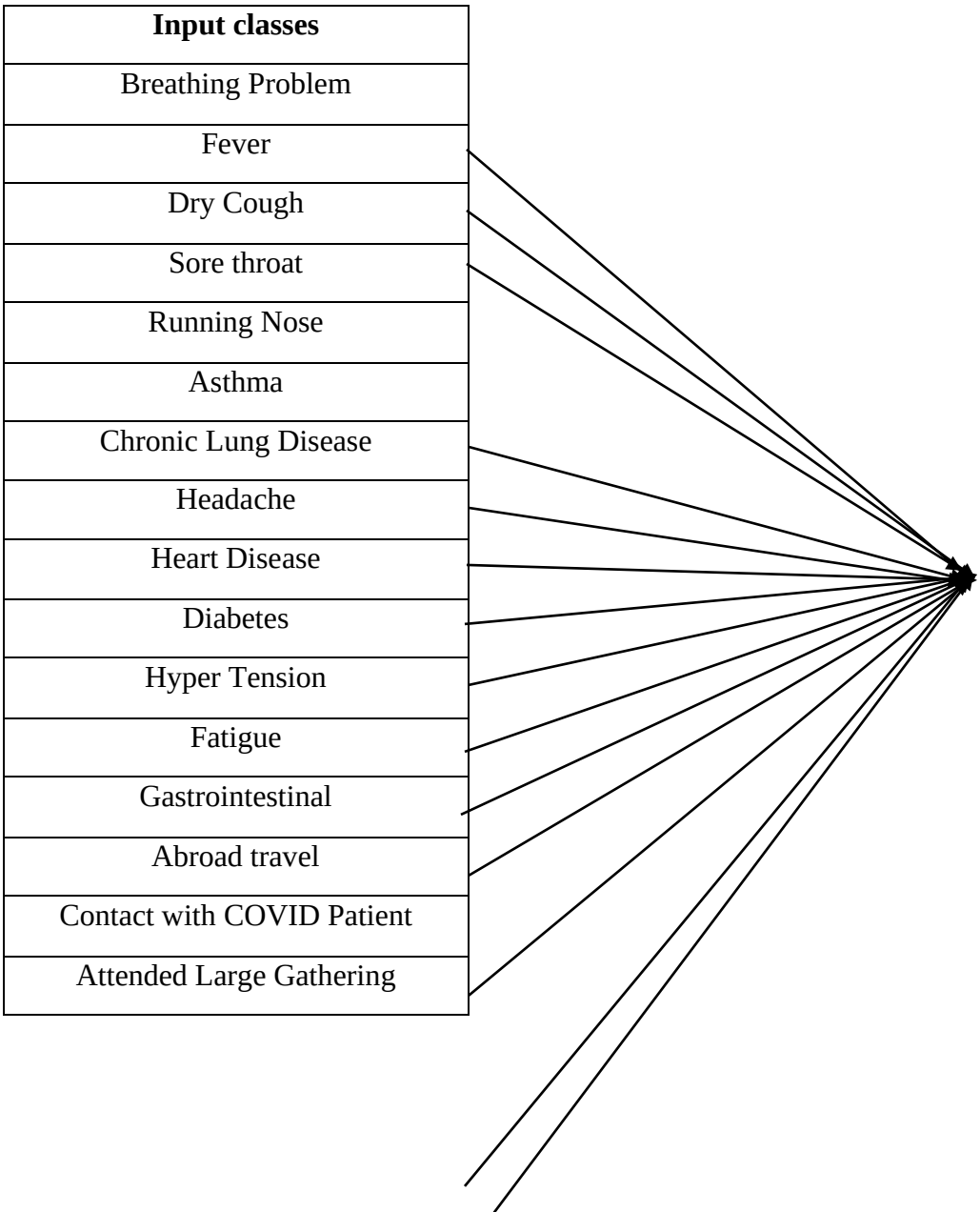
Data Pre-Processing:

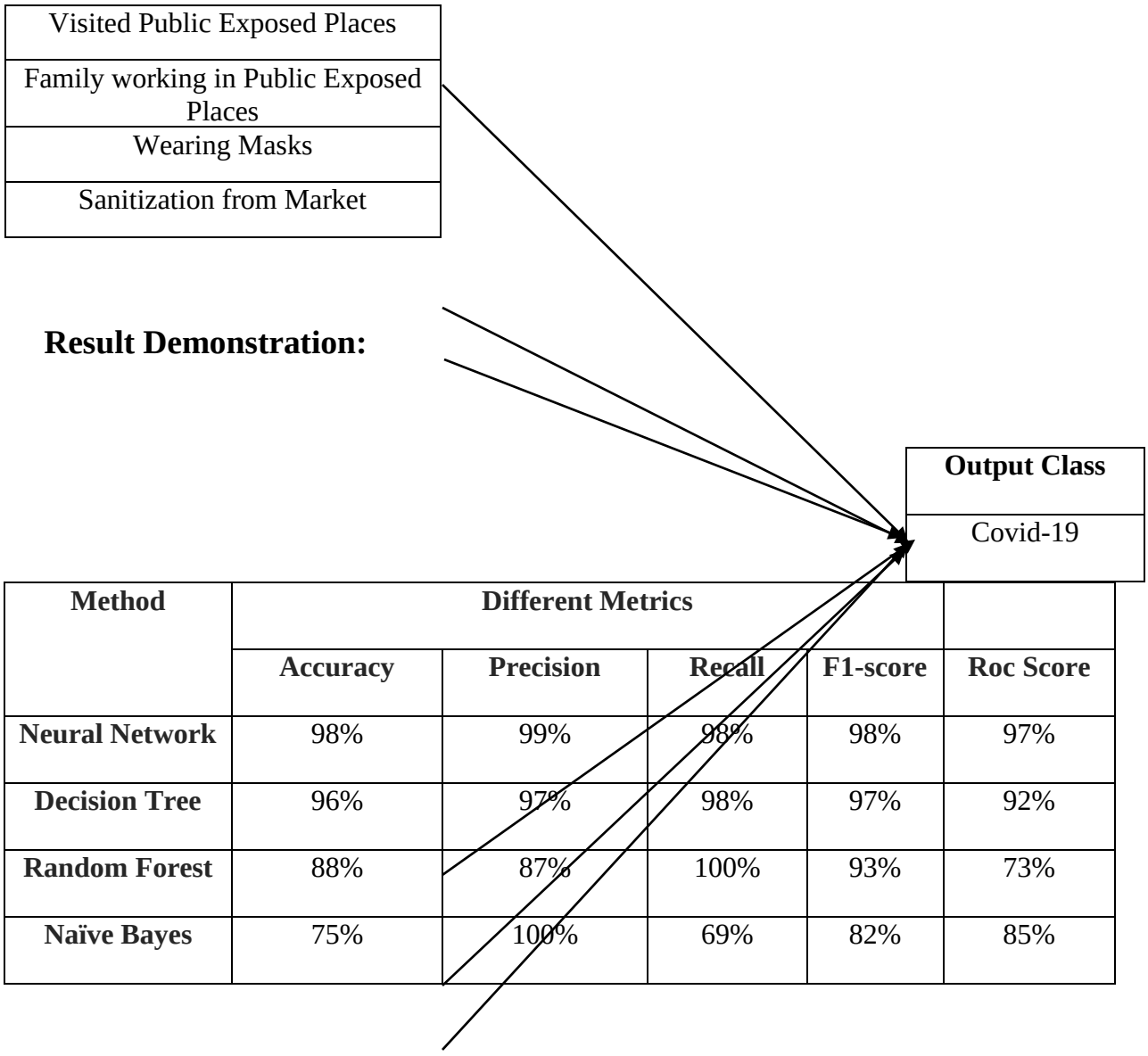
We trained and tested our method by Symptoms and COVID Presence dataset. This dataset is publicly available. There is no way to unacknowledged that a good quality dataset provides a good prediction and generalize the ability of our model. However, the dataset we choose for our experiment may not be used directly in our model because there is a lot of noise such as absence of data and unsuitable format of the dataset. Hence, preprocessing of data is essential to process the original data. Data processing and data cleaning both we have done in our project. First of all, in our dataset, it is checked that is there any column or row where exits null. If it is found, we will remove the whole row. Making suitable data for ML classification, we have converted data of each column (Yes, No) into (1, 0). After that we find out the correlation between each column with the decision factor column/class. We got two columns (Sanitization from Market and Wearing Masks) which have no correlation with decision factor class and that is why, we have dropped these two columns from our existing dataset. And we rename our decision factor class,

COVID-19 as COVID. For the training and testing purposes, we have split the total cases into 70% for training and 30% for the testing set.

Input And Output Data:

In this “Covid_Dataset” dataset there is twenty independent variables/classes and one dependent variable/class. Those twenty dependent class contains symptoms of covid disease and the class covid is the output class[29].





Result Demonstration:

Method	Different Metrics				
	Accuracy	**Precision**	**Recall**	**F1-score**	**Roc Score**
Neural Network	98%	99%	98%	98%	97%
Decision Tree	96%	97%	98%	97%	92%
Random Forest	88%	87%	100%	93%	73%
Naïve Bayes	75%	100%	69%	82%	85%

Graph Demonstration:

Roc Auc curve:

We use the AUC (Area Under The Curve) ROC (Receiver Operating Characteristics) curve to assess or display the performance of a multi-class classification task**[30]**. As AUC should be used instead of accuracy when evaluating and comparing classifiers, as AUC is a more accurate measure overall**[14]**.Using this dataset we have formed a Roc-Auc curve (figure-1).

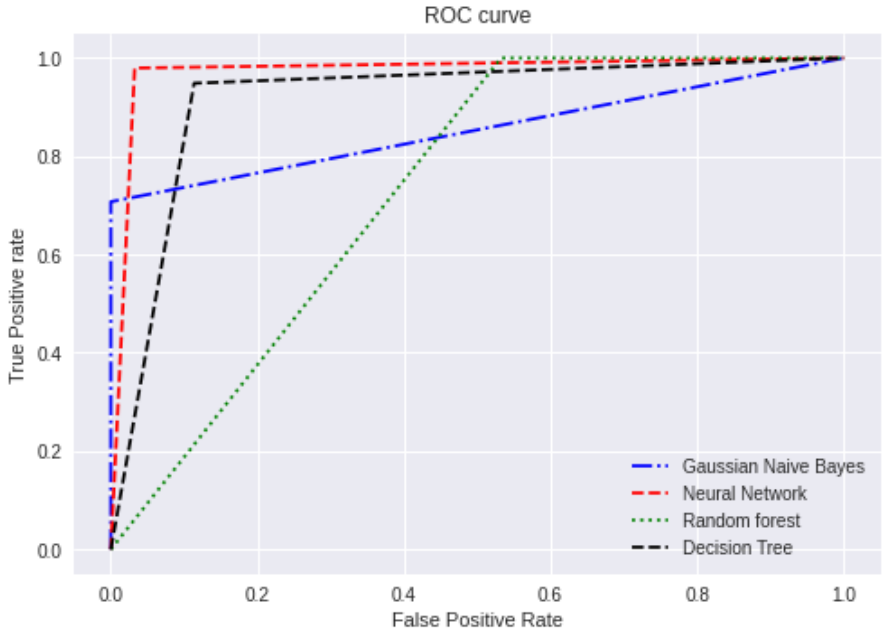


Figure 1:Roc-Auc Curve

Decision Tree:

Decision tree is logically combined sequences of basic tests in which each test compares a numeric attribute to a threshold value or a nominal attribute to a range of alternative values[16].We used Decision tree ML algorithm to predict accuracy(**Figure-2)**.

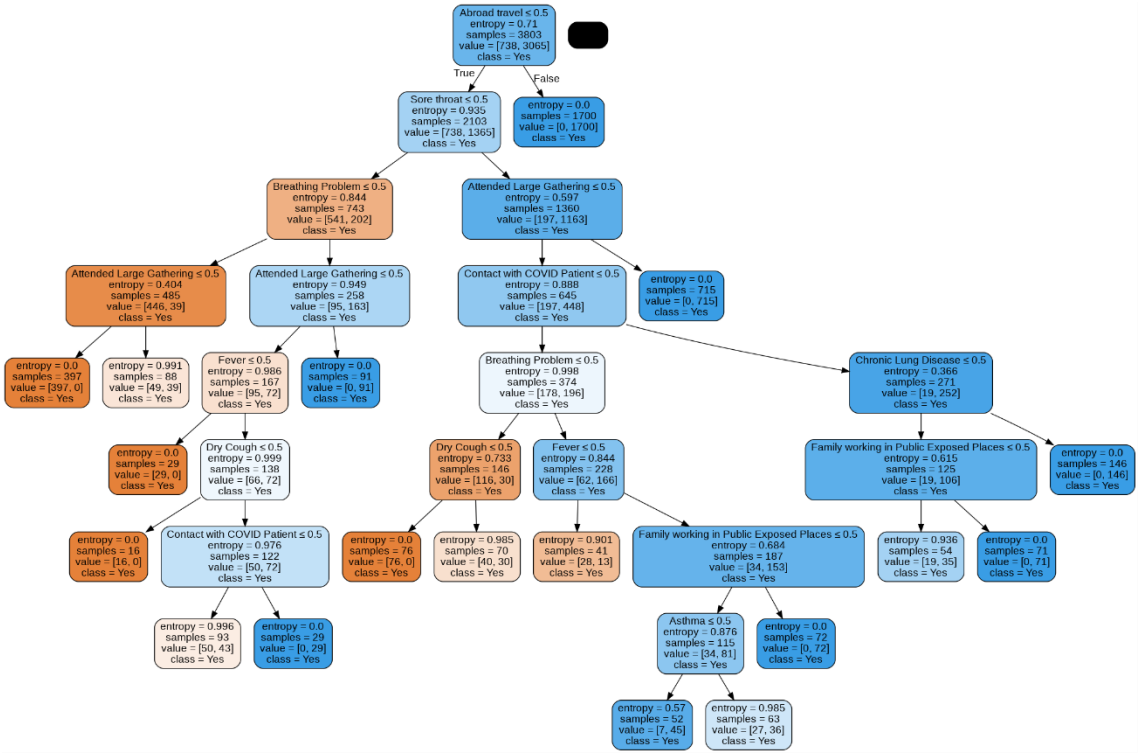


Figure 2:Decision Tree

7. Conclusion

Challenges

Dataset: Getting the perfect dataset is a challenge.

Cleaning Process: The dataset processing is a tough job. As many datasets don't have the expected value we want. Some algorithms require numeric values. So, it is necessary to clean datasets.

Deciding Factors: As there are many machine learning algorithms, it is tough job to choose one of them. It is tough to say which one will give accurate result before implementation.

Limitations

There are certain limitations in our project the main limitations we have is the dataset. As we do not have a big dataset. We cannot perform a big operation such as the spread of covid in a city. Another limitation is, there are several algorithms in machine learning where we can process our data. But we have implemented some of that. Where there is more opportunity to choose more algorithm, which can be more efficient.

Future work:

To enhance the accuracy of our prediction, we have to analyze more with COVID Symptoms dataset and others ML classification model. In future work, we will try various model which is designed for classification and besides, we will try to train and test our models for other COVID Symptoms dataset.

Conclusion

We have implemented our datasets in four different machine learning algorithms. We have seen tremendous result in some algorithm where the accuracy is close to perfection. The four algorithms we used are Gaussian Naïve Byes, Random Forest, Neural Network, Decision tree. From the implementation we have seen that Neural network has the best precision rate with 99.0%. Decision tree has the second best with a precision rate of 97.0%. Naïve Byes precision was 100% but as its accuracy level is low with only 75%, so we cannot rely on it. The accuracy percentage of Neural Network is highest amongst all. With an accuracy level of 98%.

The close behind was decision tree with 96%. So, we can say that Neural Network is the suitable amongst all. But there is a problem choosing this algorithm which is, it is best when it comes to big dataset. It works fine with the big dataset. Other than that, the accuracy level is best in Neural Network. Decision tree is better when the dataset is small compare to the dataset of Neural Network. But the accuracy of Decision tree is less than Neural Network.

So, if we want to consider the best algorithm for covid-19 detection we can say that Neural Network can detect more precisely and accurate result. Which can prevent the spread of covid-19 as people will test according to their symptoms and it will give the results. Neural network can perform better result for big datasets so it is much more suitable for a large number of populations which is a very good thing. So, we have in this paper we have not only implemented machine learning for covid detection but also discovered which machine learning algorithm is

best for this type of dataset. Although there are many algorithms to implemented which might be come with more accuracy and suitability.

8. Reference:

[1] A. S. Kwekha-Rashid, H. N. Abduljabbar, and B. Alhayani, "Coronavirus disease (COVID-19) cases analysis using machine-learning applications," *Appl. Nanosci.*, no. 0123456789, 2021, doi: 10.1007/s13204-021-01868-7.

[2] "worldmeter," *Worldometer, 'COVID Reports'. https//www.worldometers.info/coronavirus/ (accessed May 24, 2021). [2] CDC, 'Corona Symptoms'. https//www.cdc.gov/coronavirus/2019-ncov/symptoms-testing/symptoms.html (accessed May 24, 2021). [3] Kaggle, 'Dataset'. https//.*

[3] "cdc," *CDC, 'Corona Symptoms'. https//www.cdc.gov/coronavirus/2019-ncov/symptoms-testing/symptoms.html (accessed May 24, 2021).*

[4] G. National and H. Pillars, "dataset," *Kaggle, 'Dataset'. https//www.kaggle.com/hemanthhari/symptoms-and-covid-presence?fbclid=IwAR1VW3jnpkNaMeiIeqnaKqwgY6L8iPcBljB3VcP2dh5TNJsqh2HcMs-Qjqg (accessed May 23, 2021).*

[5] J. D. Novaković, A. Veljović, S. S. Ilić, Ž. Papić, and T. Milica, "Evaluation of Classification Models in Machine Learning," *Theory Appl. Math. Comput. Sci.*, vol. 7, no. 1, p. Pages: 39 – 46, 2017, [Online]. Available: https://uav.ro/applications/se/journal/index.php/TAMCS/article/view/158.

[6] F. Barboza, H. Kimura, and E. Altman, "Machine learning models and bankruptcy prediction," *Expert Syst. Appl.*, vol. 83, pp. 405–417, 2017, doi: 10.1016/j.eswa.2017.04.006.

[7] M. Saqlain, B. Jargalsaikhan, and J. Y. Lee, "A voting ensemble classifier for wafer map defect patterns identification in semiconductor manufacturing," *IEEE Trans. Semicond. Manuf.*, vol. 32, no. 2, pp. 171–182, 2019, doi: 10.1109/TSM.2019.2904306.

[8] L. Zhu, F. R. Yu, Y. Wang, B. Ning, and T. Tang, "Big Data Analytics in Action," pp. 266–294, 2018, doi: 10.4018/978-1-5225-7609-9.ch009.

[9] H. Samin and T. Azim, "Knowledge Based Recommender System for Academia Using Machine Learning: A Case Study on Higher Education Landscape of Pakistan," *IEEE Access*, vol. 7, no. c, pp. 67081–67093, 2019, doi: 10.1109/ACCESS.2019.2912012.

[10] G. S. Collins and K. G. M. Moons, "Reporting of artificial intelligence prediction models," *Lancet*, vol. 393, no. 10181, pp. 1577–1579, 2019, doi: 10.1016/S0140-6736(19)30037-6.

[11] M. A. Salam, S. Taha, and M. Ramadan, "COVID-19 detection using federated machine learning," *PLoS One*, vol. 16, no. 6 June, pp. 1–25, 2021, doi: 10.1371/journal.pone.0252573.

[12] F. Cabitza *et al.*, "Development, evaluation, and validation of machine learning models for COVID-19 detection based on routine blood tests," *Clin. Chem. Lab. Med.*, vol. 59, no. 2, pp. 421–431, 2021, doi: 10.1515/cclm-2020-1294.

[13] Y. Zoabi, S. Deri-Rozov, and N. Shomron, "Machine learning-based prediction of

COVID-19 diagnosis based on symptoms," *npj Digit. Med.*, vol. 4, no. 1, pp. 1–5, 2021, doi: 10.1038/s41746-020-00372-6.

[14] G. Quer *et al.*, "Wearable sensor data and self-reported symptoms for COVID-19 detection," *Nat. Med.*, vol. 27, no. 1, pp. 73–77, 2021, doi: 10.1038/s41591-020-1123-x.

[15] S. Agarwal, *Data mining: Data mining concepts and techniques*. 2014.

[16] S. B. Kotsiantis, "Decision trees: A recent overview," *Artif. Intell. Rev.*, vol. 39, no. 4, pp. 261–283, 2013, doi: 10.1007/s10462-011-9272-4.

[17] "Machine Learning Random Forest Algorithm," *"Machine Learn. Random For. Algorithm - Javatpoint," www.javatpoint.com. https//www.javatpoint.com/machine-learning-random-forest-algorithm (accessed Feb. 08, 2022).*

[18] "RF," *"Introduction to Random For. Mach. Learn. Eng. Educ. Progr. | Sect. https//www.section.io/engineering-education/introduction-to-random-forest-in-machine-learning/ (accessed Feb. 08, 2022).*

[19] T. Windeatt, "Ensemble MLP classifier design," *Stud. Comput. Intell.*, vol. 137, no. 5, pp. 133–147, 2008, doi: 10.1007/978-3-540-79474-5_6.

[20] M. A. Sovierzoski, F. I. M. Argoud, and F. M. De Azevedo, "Evaluation of ANN classifiers during supervised training with ROC Analysis and Cross Validation," *Biomed. Eng. Informatics New Dev. Futur. - Proc. 1st Int. Conf. Biomed. Eng. Informatics, BMEI 2008*, vol. 1, pp. 274–278, 2008, doi: 10.1109/BMEI.2008.251.

[21] Y. Kutlu, M. Kuntalp, and D. Kuntalp, "Optimizing the performance of an MLP classifier for the automatic detection of epileptic spikes," *Expert Syst. Appl.*, vol. 36, no. 4, pp.

7567–7575, 2009, doi: 10.1016/j.eswa.2008.09.052.

[22] M. Kubat, R. C. Holte, and S. Matwin, “Machine learning for the detection of oil spills in satellite radar images,” *Mach. Learn.*, vol. 30, no. 2–3, pp. 195–215, 1998, doi: 10.1023/a:1007452223027.

[23] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, “Recommendation systems: Principles, methods and evaluation,” *Egypt. Informatics J.*, vol. 16, no. 3, pp. 261–273, 2015, doi: 10.1016/j.eij.2015.06.005.

[24] “Edited by Sofia Visa, Atsushi Inoue, and Anca Ralescu,” 2011.

[25] F. Spoto, P. Martimort, and M. Drusch, “Sentinel - 2: ESA’s optical high-resolution mission for GMES operational services,” *Eur. Sp. Agency, (Special Publ. ESA SP*, vol. 707 SP, 2012.

[26] H. C. Sox, M. C. Higgins, and D. K. Owens, “Measuring the Accuracy of Diagnostic Information,” *Med. Decis. Mak.*, pp. 93–142, 2013, doi: 10.1002/9781118341544.ch5.

[27] F. Kohavi, R. & Provost, “Glossary of terms: Machine Learning,” *Mach. Learn.*, vol. 30, pp. 271–274, 1998, [Online]. Available: http://ai.stanford.edu/~ronnyk/glossary.html%0Ahttp://robotics.stanford.edu/~ronnyk/glossary.html.

[28] F. Provost and T. Fawcett, “Analysis Comparison and Visualization under Imprecise,” *Third Internat. Conf. Knowl. Discov. Data Min.*, pp. 43–48, 1997.

[29] “covid symptoms,” *https://www.cdc.gov/coronavirus/2019-ncov/symptoms-testing/symptoms.html*.

[30] Sarang Narkhede, “Understanding AUC - ROC Curve,” *Towar. Data Sci.*, pp. 6–11, 2019, [Online]. Available: https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5.