Machine learning approach for identification of SARS-CoV-2 variants

Gazi Mahfuzur Rahman

Id: 2014-3-60-027

Khandokar Zaeem Hasan

Id: 2016-1-68-016

Habibur Rahman Reyad

Id: 2015-2-60-042

A thesis submitted in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering

Department of Computer Science and
Engineering East West University

Dhaka-1212, Bangladesh

July 2021

Declaration

We, hereby, declare that the work presented in this thesis is the outcome of the investigation performed by us under the supervision of Dr. Taskeed Jabid, Chairperson, Associate Professor, Department of Computer Science and Engineering from East West University. We also declare that no part of this thesis has been or is being submitted elsewhere for the award of any degree or diploma.

Counter signature

…………………
(Taskeed Jabid)
Supervisor

Signature

…………………
(Gazi Mahfuzur Rahman)
(2014-3-60-027)

Signature

…………………
(Khandokar Zaeem Hasan)
(2016-1-68-016)

Signature

…………………
(Habibur Rahman Reyad)
(2015-2-60-042)

Letter of Acceptance

This thesis report entitled ***“Machine learning approach for identification of SARS Cov-2 variant"*** submitted by Gazi Mahfuzur Rahman (ID: 2014-2-60-027), Khandokar Zaeem Hasan (ID: 2016-1-68-016), and Habibur Rahman Reyad (ID: 2015-2-60-042) to the Department of Computer Science and Engineering, East West University is accepted by the department in partial fulfillment of requirements for the award of the Degree of Bachelor of Science and Engineering on July 2021.

Supervisor

....

(Dr. Taskeed Jabid)

Chairperson and Associate Professor,

Department of Computer Science and Engineering, East West University.

Chairperson

....

(Dr. Taskeed Jabid)

Chairperson and Associate Professor,

Department of Computer Science and Engineering, East West University

Abstract

To finding SARS-CoV-2 variants is not a trivial task. In 2019, coronavirus was spreading all over the world. Where scientists are finding different kind of coronavirus variants around the world. In this research paper, we are detecting the variants by Machine learning approach. For the purpose, we collected our data from GISAID. We used 5 algorithms. They are Logistic regression, K-Nearest Neighbor, Artificial Neural Network, Convolutional Neural Network and Support vector machine. From those, models we are comparing the f1 macro and accuracy. From the analysis Convolutional neural network performs well at variants detection

Acknowledgment

First of all, thanks to Almighty Allah. We are conveying our gratitude to the honorable thesis supervisor, Dr. Taskeen Jabed, Chairperson and Associate Professor of the Department of Computer Science and Engineering, East West University. He allowed us to work with him on the topic of “***Machine learning approach for identification of SARS Cov-2 variant***”. Without the help from him, continuous support, assessments, and guidance it wouldn’t be possible for us to complete this thesis alone.

Gazi Mahfuzur Rahman
Khandokar Zaeem Hasan
Habibur Rahman Reyad
July 2021

Flow Chart

Chapter 1

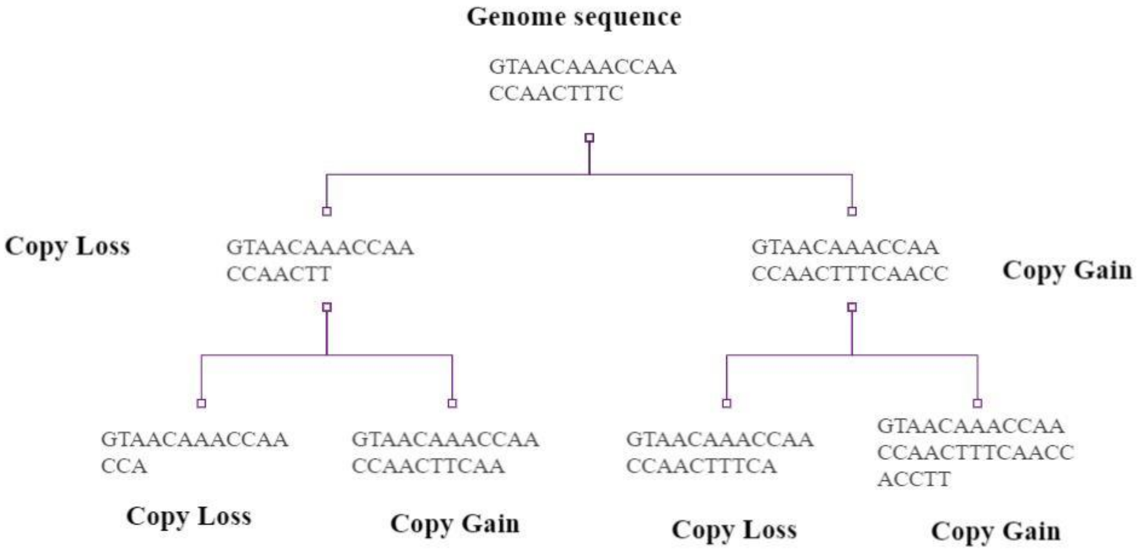
1.1 Introduction

From 2003 to 2014 there were 3 important viruses found in our world. Before the novel Coronavirus, those viruses infected many people. The first virus was found in Asia in 2003 known as SARS-CoV-1(severe acute respiratory syndrome)[1]. After that, MERS (Middle East respiratory syndrome) was found in Saudi Arabia in 2012[2]. In 2019 the world knows about a new virus that can spread from infected person to disinfected person. The infected person is breathing out droplets and small particles which contain that virus. Generally, the droplets and small particles come out by sneeze, coughing. Those small particles and droplets are landing on the eyes, noses, and mouth. It also spreads by touching surfaces where viruses are. The doctors are saying if anyone is within 6 feet distance from an infected person then others are most probably infected[3]. However, day by day the cases are more serious. On November 5, 2020, the USA government lockdown the area so that the infection was decreasing. But in KENT near London, the infection cases are increasing. As a result, there was a disaster.

Chapter 1

1.2 Variants of SARS-CoV-2

The novel coronavirus is found in every country of the world. The characteristic of the virus is different apart from the common characteristic. So, there is confusion about the mutation of the virus. Then there is a term known as Variants. Viruses are multiplied by copying their genomes over and over, like old photocopies, these copies aren't always perfect i.e., If we coy a paper for multiple times then the copies are not same all the time. First copies of the photocopy of the paper were good but after that the last copies are not have same quality with respect to the first copies. Each of these imperfect copies is known as Variants. There are some gains and losses from the genome sequence of the original copy of coronavirus. Also, the base of the genome can be replaced by genome sequences[4].



In recent times there are almost a bunch of variants. Those variants are found in almost many countries. But the source is mentioned in this section[5].

Country/Region	Scientific name	WHO name
UK(Kent)	B.1.1.7	Alpha
South Africa	B.1.351	Beta
Brazil	P.1	Gemma
India	B.1.617.2	Delta

In this report paper, we are trying to find out the variants by the machine learning approach.

They named the mutation B.1.1.7. The new mutation was much more transmissible, spreading fast. Two months later it was spreading to 30 countries around the world. After 5 months the virus was a common variation in the USA. Lately, there are more and more variants.

The virus is just a shell of protein which is surrounding some genetic material such as DNA or RNA. That genetic material is a bunch of molecules that can be represented as a series of letters (A, G, C, T). These is creating the genome sequence of the SARS CoV-2. The sequence is >29000 bp long. The slight change of nitrogen base A, C, G, T can different from the original copies i.e., Let assume the genome sequence of the SARS is,

AAACGTTTACCCATTTCGG**CCGCC**GATTTCCCAAACCCGG.
AAACGTTTACCCATTTCGG**GCGGC**GATGTTTCCCAAACCCGG.

This change can create new variants from the old genome sequence.

Chapter 1

1.3 Variants in Bangladesh

In March 8,2020 Bangladesh confirmed that coronavirus was found in the country for the first time. After the flight from Italy, some people are tested positive.

Since November of last year, Bangladesh has confirmed the presence of five Covid 19 variations, the most recent of which being the Indian version (B.1.617.2), The UK (B.1.1.7), South Africa (B.1.351), Nigeria (B.1.525), and Brazilian (P1) coronavirus strains have also been detected in the nation. The UK variation was discovered in January, although it was only confirmed in March. By late April, three more variants had been confirmed. Prof Dr Tahmina Shirin, Head of the Department of Epidemiology Disease Control and Research (IEDCR), said the Indian strain (B.1.617.2) had been detected in six samples collected between April 27 and 30[6].

The variation (B.1.617.2) is among three subgroups of the Indian variant that have yet to be discovered (B.1.617). Five of the six genomes sequenced have been uploaded to GISAID, a global science project that provides open-access to influenza and coronavirus genetic data. However, the IEDCR announced on Sunday that it had discovered four variations in 200 samples after sequencing them. The first variant was found from a sample taken on April 27, according to outbreak.info, a Covid variant-tracking website based on GISAID. 1,466 genome sequences already submitted into the website. According to the variation-tracking website Outbreak.info, after genome sequences of samples gathered since the variant was originally discovered in the country were completed, 38 percent of them were confirmed to be South Asian variants. The UK variant accounts for around 14% of them, according to the report, which also stated that the exact ratio of the Indian variant has yet to be determined. According to Outbreak.info, the prevalence of the UK variant is higher in Sylhet, where 19 of the genome sequences were discovered. However, the variation was discovered in 40 of the 809 samples collected in Dhaka. In Chittagong, eight out of 189 samples have UK variations. The South African variation has been found in at least ten districts across the country, with Dhaka accounting for the most cases (215 out of 809 genome sequences). Nigerian variations were discovered in samples gathered from two divisions, according to IEDCR sources. Analyses of all the samples sequenced revealed that 37 percent of these variants carried the S:E484K mutation, which is known to make patients more contagious and lethal. The absence of the mutation in the Indian variety identified in Bangladesh, according to the health officials, is a relief for Bangladesh. It does, however, occur in South African, Nigerian, and Brazilian variations. According to the GISAID website, the Bangladesh Council of Scientific and Industrial Research (BCSIR) posted on April 20 the only Brazilian variant genome sequence (P1) that was taken from a sample received from a female patient in the capital on February 18[7].

Chapter 2

Literature Review

SARS-CoV and SARS-CoV-2, these two variants differed [8]. Using molecular dynamics (MD) simulations, Machine learning (ML), and free-energy perturbation (FEP) calculations, they explicate the differences in binding by the two viruses, where ML performs distinctly better than MD and FEP. Besides Nguyen, Ngoc Giang Tran et al [9] convolutional neural network (CNN) has applied to identify Deoxyribonucleic Acid (DNA) sequence. In their research they have used 12 DNA sequence datasets and evaluated with model and obtained significant results. Siquan et al [10] tries to identify DNA binding protein from amino acid which is not a trivial task. But computational methods like SVM, DNABP and CNN-RNN, are not efficient methods. Therefore, they proposed coordinates a bidirectional long-term memory recurrent neural network and a convolutional neural network, called CNN-BiLSTM. This model identifies where DNA binding proteins, where CNN-BiLSTM returns higher accuracy and proved to be more efficient for identifying DNA-binding proteins. Lopez-Rincon et al [11] used CNN classifier used to train genomic sequences data of SARS-CoV-2. The results of the experiment later synthesized by National Center for Biotechnology Information (ncbi) and Global Initiative. The model is able to differentiate SARS-CoV-2 from different virus strains with desired accuracy. Lopez-Rincon et al [12], they proposed deep learning methodology to obtained primer collections for every independent variant. This method can be helpful for preliminary diagnosis for COVID-19 affected patients. They also proposed a methodology that can be applied to track COVID-19 patients. As well as, it is able to differentiate different variants.

Chapter 3

Data Preprocessing

In this COVID-19 pandemic situation, we can see that the virus has mutated and become a more threatening virus towards us. Day by day it's evolving and from the Centers for Disease Control and Prevention (CDC), we found that until now there are many types of variants that have evolved. The criteria are Alpha (B.1.1.7), Beta (B.1.351), Gamma (P.1), and Delta (B.1.617.2). These variants are circulating in the whole world. The recent delta variant is far more dangerous than the other variant. We conduct research that is based on machine learning. With the help of machine learning, we can easily detect these multi variates among the human body. It will also help us to take the necessary precautions.

To conduct our research, we need data. We have collected the data from the “GISAID” database which is an online-based data center that provides open access to genomic data of influenza viruses and the coronavirus responsible for the COVID-19 pandemic. The genome sequence is > 29000 bp long[13]. The genome sequence consists of “ACTG” also there are some additional nucleotides that are very few in numbers.

First, we have to understand the data and how many null values are there, and how the nucleotide forms. The downloaded data is in FASTA format, with help of the python package (Bio Python) we can convert this data to CSV format which makes it convenient to read the file in a panda’s data frame and also makes the work process easy in python. Next, we convert the string data to vector format as a prerequisite for running the selected machine learning algorithm. So that our machine can read and fit our desired model to check which model is best for variant classification. Before fitting the model, we need to clean and prepare the data. From “GISAID” we can find a large number of datasets. We selected the recent data of covid-19, a total of 8000 genome sequences of nucleotides. Nucleic acids are made up of nucleotides, which are the basic building blocks. Long chains of nucleotides make up RNA and DNA, which are polymers. A sugar molecule (ribose in RNA or deoxyribose in DNA) is bonded to a phosphate group and a nitrogen-containing base to form a nucleotide. Adenine (A), cytosine (C), guanine (G), and thymine (T) are the four bases used in DNA. The nucleotide uracil (U) replaces thymine in RNA[14]. Our data is based on DNA. We know that the four native bases for DNA are AGTC, however, some of the sequences contain the letter 'N', which illustrates that these nucleotide bases are not deciphered correctly, thus it is an unidentified nucleotide. So, N cannot be replaced with any other base until we have chromatogram data that can fix the bases in case of ambiguity or remove the “N” data[15]. For simplicity, we convert the data into vector format as, N: 0, A: 4, G: 3, C: 1, T: 2, and others are chosen randomly from the given Table 1[16].

Nucleic Acid Code	Meaning	Mnemonic
A	A	Adenine
C	C	Cytokine
G	G	Guanine
T	T	Thymine
U	U	Uracil
R	A or G	purine
Y	C, T, or U	pyrimidines
K	G, T, or U	bases which are Ketones
M	A, or C	bases with amino groups
S	C, or G	Strong interaction
W	A, T, or U	Weak interaction
B	not A (i.e., C, G, T, or U)	B comes after A

D	not C (i.e., A, G, T, or U)	D comes after C
H	not G (i.e., A, C, T, or U)	H comes after G
V	neither T nor U (i.e., A, C, or G)	V comes after U
N	A C G T U	Nucleic acid

Table 1: Sequence Representation

Chapter 4

Methodology

The capacity of the machine to assign instances to their correct groupings is referred to as classification in artificial intelligence and machine learning. Computer vision also can classify i.e., a machine can determine whether a picture comprises a cat or a dog, or whether or not an image contains a human body. Natural Language Processing (NLP) allows a computer to determine if a sentence is good, negative, or neutral. The machine must learn the patterns of that assignment from the training features provided in a labeled training data set before it can decide how to allocate an instance to its group. So as for this variant classification, we apply different machine learning algorithms to see which gives the best result.

Chapter 4

4.1 Linear Regression

Linear Regression is a supervised Machine Learning model that determines the best fit linear line between the independent and dependent variables, or the linear connection between the two variables. There are two forms of linear regression: simple and multiple[17]. Only one independent variable is present in simple linear regression, and the model must identify a linear relationship between it and the dependent variable. Multiple Linear Regression, on the other hand, uses more than one independent variable to find a relationship. The basic goal of a Linear Regression model is to determine the best-fit linear line and the appropriate intercept and coefficient values such that the error is minimized. The discrepancy between the actual and predicted values is called error, and the goal is to reduce it.

First, we run a simple linear regression on each variant, there are 4 variants, so we take 6 combinations of 4 variants i.e., alpha against beta, alpha against gamma, alpha against delta, beta against gamma, beta against delta, and gamma against delta to apply 6 models. For this purpose, we take 1500 data sets for each variant e.g., 1500 for alpha and 1500 for beta, and so on. We trained 75% of the data and 25% data for testing purposes. The prediction score is shown in Table 2. In this pandemic situation, we need a more reliable model to predict the 4 variants together.

Class	**Linear Regression**
Alpha against Beta	69.7%
Alpha against Gamma	88.6%
Alpha against Delta	96.5%
Beta against Gamma	82.0%
Beta against Delta	94.0%
Gamma against Delta	96.4%

Table 2: Linear Regression Classification Accuracy

Chapter 4

4.2 Logistic Regression

The logistic function, which is at the heart of the procedure, is called logistic regression[18]. The logistic function, also known as the sigmoid function, was created by statisticians to characterize the properties of population increase in ecology, such as how it rises swiftly and eventually reaches the environment's carrying capacity. It's an S-shaped curve that can transfer any real-valued integer to a value between 0 and 1 but never exactly between those two points. Logistic regression, like linear regression, uses an equation as its representation. To anticipate an output value, input values are blended linearly using weights or coefficient values. The output value being modeled is a binary value (0 or 1) rather than a numeric value, which is a fundamental difference from linear regression. The probability of the default class is modeled using logistic regression (e.g., the first class). For instance, if we are modeling two variants as alpha or beta from genome sequences, then the first-class could be alpha and the logistic regression model could be written as the probability of alpha given a variant’s sequence, or more formally:

$$P\ (\text{variant}=\text{alpha} \mid \text{sequence})$$

The logistic regression algorithm's coefficients must be computed using our training data. A maximum-likelihood estimate is used for this. Although it makes assumptions about the distribution of our data, maximum-likelihood estimation is a typical learning process utilized by several machine learning algorithms. The optimal coefficients would result in a model that predicted a value very close to 1 (alpha) for the default class and a value very close to 0 (beta) for the other class for the default class. The idea behind maximum-likelihood logistic regression is that a search technique looks for coefficients (Beta values) that minimize the difference between the model's predicted probabilities and those in the data (e.g. probability of 1 if the data is the primary class). It's as simple as plugging numbers into the logistic regression equation and calculating a result to make predictions with a logistic regression model. There are 4 variants, so we take 6 combinations of 4 variants i.e., alpha against beta, alpha against gamma, alpha against delta, beta against gamma, beta against delta, and gamma against delta to apply 6 models. For this purpose, we take 1500 data sets for each variant e.g., 1500 for alpha and 1500 for beta, and so on. We trained 75% of the data and 25% data for testing purposes, this dataset is only for binary classification. The prediction score is shown in Table 3.

Class	**Logistic Regression**
Alpha against Beta	95.2%
Alpha against Gamma	97.7%
Alpha against Delta	99.0%
Beta against Gamma	97.4%
Beta against Delta	98.4%
Gamma against Delta	99.0%

Table 3: Logistic Regression Classification Accuracy

Further, we apply a modified version of logistic regression that predicts a multinomial probability (i.e., more than two classes) for each input sample. For multi-class classification, we select 6000 data samples. From this sample, we train 80% and the rest 20% for validation (fixed random state) i.e., means we train 4800 data samples and 1200 for validation purposes also we test our model with 2000 data samples. The validation score is 93.4%, and the prediction score on test data is 93.3%. We also cross-validated our model to check the accuracy and the f1-score. We cross-validate 5 items and take the average value of them. Model precision, recall, f1-score, and the support shown in Table 4. Further accuracy and f1-macro of validation and test data score with cross-validation are shown in Table 5.

Class	**Precision**	**Recall**	**F1-score**	**Support**
Alpha	0.91	0.93	0.92	500
Beta	0.91	0.91	0.91	500
Gamma	0.93	0.94	0.94	500
Delta	0.99	0.95	0.97	500
Macro Avg	0.93	0.93	0.93	2000

Table 4: Classification Report of Logistic Regression

	Validation Score	**Test Score**	**Average**
Accuracy	93.4%	93.3%	93.3%
Cross-validation=5 (accuracy)	91.0%	91.9%	91.5%
Cross-validation=5 (f1_macro)	91.0%	91.9%	91.5%

Table 5: Final Score of Logistic Regression

Chapter 4

4.3 K-Nearest Neighbors

K-Nearest Neighbors (KNN) [19] is one of the most basic supervised algorithms for regression and classification problems in Machine Learning. KNN algorithms take data and apply similarity metrics to classify fresh data points (e.g., Distance function). A majority vote is used to classify its neighbors. One major advantage of this algorithm is that there’s no need to build a model, tune several parameters, or make additional assumptions. And one major disadvantage is that the algorithm gets significantly slower as the number of example variables increases. We train 80%, and 20% for validation i.e., 4800 samples of data for training, and 1200 samples of data for validation purposes also we test our model with 2000 data samples.

We choose the Minkowski distance metric for our classification problem to calculate the distance between test samples and trained samples. The Minkowski distance, often known as the Minkowski metric, is a metric in a normed vector space that is a generalization of the Euclidean and Manhattan distances. The validation score is 96.4%, and the prediction score on test data is 97.6%. We also cross-validated our model to check the accuracy and the f1-score. We cross-validate 5 items and take the average value of them. Model precision, recall, f1-score, and the support shown in Table 7. Further accuracy and f1-macro of validation and test data score with cross-validation are shown in Table 8.

Class	**Precision**	**Recall**	**F1-score**	**Support**
Alpha	0.98	0.97	0.98	500

Beta	0.97	0.96	0.97	500
Gamma	0.97	0.98	0.98	500
Delta	0.99	0.98	0.98	500
Macro Avg	0.98	0.98	0.98	2000

Table 7: Classification Report of K-Nearest Neighbors

	Validation Score	**Test Score**	**Average**
Accuracy	96.4%	97.6%	97.0%
Cross-validation=5 (accuracy)	91.6%	94.4%	93.0%
Cross-validation=5 (f1_macro)	91.6%	94.4%	93.0%

Table 8: Final Score of K-Nearest Neighbors

Chapter 4

4.4 Support Vector Machine

SVM is a supervised machine learning technique that may be used to help with classification and regression difficulties[20]. Its goal is to discover the best compromise between the many outputs. Simply, SVM performs sophisticated data transformations based on the kernel function we choose and then seeks to optimize the separation boundaries between our data points based on the labels or classes we designate.

SVM seeks to find a line that optimizes the distance between a two-class data set of 2-dimensional space points in its most basic form, linear separation. To generalize, the goal is to find a hyperplane in n-dimensional space that maximizes the distance of data points from their perspective classes. Support Vectors are the data points with the shortest distance to the hyperplane (closest points).

A kernel function is used to compute the separation of data points. Linear, Polynomial, Radial Basis Function (RBF), and Sigmoid are the kernel functions. Simply put, the smoothness and efficiency of class separation are determined by these functions, and experimenting with their hyperparameters can lead to overfitting or underfitting.

In our model, we experiment with different kernel functions i.e., linear, polynomial, rbf (radial basis function) to check which function classifies the best. We train 80%, and 20% for validation i.e., 4800 samples of data for training, and 1200 samples of data for validation purposes also we test our model with 2000 data samples. We apply this method in all the different kernel functions. The validation accuracy of the “linear”, “rbf”, and “polynomial” function is 95.7%, 93.8%, and 96.6% respectively, and test accuracy of the “linear”, “rbf”, and “polynomial” function is 96.2%, 94.7%, and 97.0% respectively. Further, we cross-validated our model to check the accuracy and the f1-score. We cross-validate 5 items and take the average value of them, which are shown in Table 9.

Data		**SVM (kernel= “linear”)**	**SVM (kernel= “rbf”)**	**SVM (kernel= “poly”)**
Validation	**Validation Accuracy**	95.9%	93.7%	96.3%
	Cross-validation=5 (accuracy)	91.3%	87.3%	91.5%
	Cross-validation=5 (f1_macro)	91.4%	87.6%	91.6%
Test	**Test Accuracy**	96.7%	95.1%	97.1%
	Cross-validation=5 (accuracy)	92.8%	89.3%	93.8%

	Cross-validation=5 (f1_macro)	92.8%	89.4%	93.8%

Table 9: Accuracy and cross-validation of SVM (kernel= “linear”, “rbf”, “poly”)

We average all three function scores i.e., validation score and test score, furthermore compare them with each other shown in Table 10. From the analysis, we see that the kernel= “poly” score is greater than the other scores. So we choose the polynomial function for this model[21]. Model precision, recall, f1-score, and the support are shown in Table 11.

Kernel	**Accuracy**	**F1 _macro**
SVM(kernel= “linear”)	92.0%	92.1%
SVM(kernel= “rbf”)	88.3%	88.5%
SVM(kernel= “poly”)	92.7%	92.7%

Table 10: Final Score of Support Vector Machine

Class	**Precision**	**Recall**	**F1-score**	**Support**
Alpha	0.95	0.97	0.96	500
Beta	0.96	0.97	0.97	500
Gamma	0.98	0.97	0.97	500
Delta	0.99	0.98	0.99	500
Macro Avg	0.97	0.97	0.97	2000

Table 11: Classification Report of Support Vector Machine

Chapter 4

4.5 Artificial Neural Network

Artificial neural networks (ANNs), also known as neural networks (NNs), its computing systems inspired by the biological neural networks that constitute human brains. It accepts some data as input and assigns a random weight to it. To optimize the weight and find the best accuracy, the number of hidden layers is responsible for this[22].

In ANN there are some hyper-parameters that we need to adjust to achieve our desired model. These hyper-parameters are learning rate, number of layers, number of neurons, regularized, etc. So to get our desired model for the best accuracy we tweak all these hyper-parameters to check which model gives us the highest accuracy and f1-macro.

We train 80%, and 20% for validation i.e., 4800 samples of data for training, and 1200 samples of data for validation purposes also we test our model with 2000 data samples. In ANN we use 4 hidden layers, each layer consisting of 16 neurons and RELU as an activation function. To mitigate the overfitting, we set the regularized to 0.0001, and to get the optimal value we set the learning rate to 0.0001. For the output layer, we have 4 neurons as there are only 4 variants to classify and we use SoftMax as an activation function because we are classifying multi-class labels.

RELU (Rectified linear activation function) works for rectifying the value if the value is positive then it will output the value as it is otherwise it will generate zero. We use Adam as our optimizer. Adam is capable of handling noisy data. For the loss function, we use sparse categorical cross-entropy. SoftMax is a mathematical function that turns a vector of integers into a vector of probabilities, with each value's probability proportional to its relative scale in the vector. Model precision, recall, f1-score, support, accuracy, and macro avg are shown in Table 12. Model scores are shown in Table 13.

Class	**Precision**	**Recall**	**F1-score**	**Sample**	**Accuracy**
Alpha	0.85	0.93	0.88	500	91.4%
Beta	0.94	0.84	0.89	500	
Gamma	0.90	0.95	0.93	500	
Delta	0.98	0.93	0.96	500	
Macro avg	0.92	0.91	0.91	2000	

Table 12: Classification Report of ANN

	Accuracy	**F1-macro**
Score	91.4%	91.0%

Table 13: Score of ANN

Chapter 4

4.5 Convolutional Neural Network

A convolutional neural network (CNN) is a sort of artificial neural network that analyzes data using perceptron's, which are machine learning unit algorithms. CNN's most important feature is that it can detect unique patterns from the given data samples. CNN's have hidden layers called convolutional layers, also there are other non-convolutional layers as well, but the basis of a CNN is the convolutional layers. A convolutional layer receives input, transforms the input in some way, and then outputs the transformed input to the next layer. With a convolutional layer, the transformation that occurs is called a convolution operation.

Convolution is the first layer to extract features from the given datasets. Convolution preserves the relationship by learning the features from input datasets.

For our purpose we use conv1d [23]. Basically, conv1D was designed for a sequence analysis - to have convolutional filters which would be the same no matter in which part of a sequence we are. The second dimension is the dimension of the so-called features where we could have a vector of multiple features at each of the timesteps.

We train 80%, and 20% for validation i.e., 4800 samples of data for training, and 1200 samples of data for validation purposes also we test our model with 2000 data samples. In CNN (conv1D) we use 2 hidden layers, each layer consisting of 16 neurons and RELU as an activation function. To mitigate the overfitting we set the l2 regularizer [24] to 0.0001, and to get the optimal value we set the learning rate to 0.0001. In the hidden layer, the kernel size is set to 21. We use max pooling [25], strides 1, and pool size is 3. Pooling mainly helps in extracting sharp and smooth features. It is also done to reduce variance, computations, and dimensionality. Also, max-pooling helps in extracting low-level features like edges, points, etc. This filter is slid over the input to conduct the convolution action. We execute element-by-element matrix multiplication and total the results at each position. This total is used to create a feature map. For the output layer, we have 4 neurons as there are only 4 variants to classify and we use softmax as an activation function because we are classifying multi-class labels.

RELU (Rectified linear activation function) works for rectifying the value if the value is positive then it will output the value as it is otherwise it will generate zero. We use Adam as our optimizer. Adam is capable of handling noisy data. For the loss function, we use sparse categorical cross-entropy. Softmax is a mathematical function that turns a vector of integers into a vector of probabilities, with each value's probability proportional to its relative scale in the vector. Model precision, recall, f1-score, support, accuracy, and macro avg are shown in Table 14. Model scores are shown in Table 15.

Class	**Precision**	**Recall**	**F1-score**	**Sample**	**Accuracy**
Alpha	0.99	0.99	0.99	500	98.6%
Beta	0.96	0.99	0.97	500	
Gamma	0.99	0.99	0.99	500	

Delta	0.99	0.98	0.98	500	
Macro avg	0.99	0.98	0.99	2000	

Table 14: Classification Report of CNN

	Accuracy	**F1-macro**
Score	98.9%	99.0%

Table 15: Score of ANN

Chapter 4

Analysis

The SARS-CoV-2 virus is mutated day by day and it's getting more dangerous to us human beings. From the nature of our data, we see that the more undefined bases are the harder it is to classify the variant in the analysis. Detecting the variant without the help of machine learning will be a tough task to do. Even a slight change in the genome sequence makes it harder.

Throughout the process, we came to see that if there are too many null values in the dataset then the model can't accurately classify the variant. So, we take the high and low coverage data from GISAID. According to GISAID "high coverage"[26] means, "only entries with <1% Ns (undefined bases) and < 0.05% unique amino acid mutations (not seen in other sequences in the database), and "complete" means, GISAID considers genomes >29,000bp[13] as complete and further assigns labels of high coverage <1% Ns (undefined bases) and low coverage >5% Ns[27].

We conduct different machine learning algorithms to check which algorithm works well on variant classification. From the methodologies section, we see that these algorithms did well on classification. Though our target is to correctly classify the variants. As we see the number of covid-19 cases is going as well as the mutation of the viruses. The mutations are stronger and more dangerous compared to other previous variants. So, we want the false positive and the false negative to be as lower as possible. Suppose, we can consider that higher recall is more important than getting higher accuracy - we want to identify as many cancer patients as possible. In the case of some other models, such as classifying whether a bank customer is a loan defaulter, it is desirable to have a high precision since the bank wouldn't want to lose customers who were denied a loan based on the model's prediction that they would be defaulters. Again, if the physician informs us that the patients who are suffering from heart disease have been classified incorrectly because they may be indicative of any other disease, then we aim not only for higher recall but also for higher precision. So, we use the f1-score which is the harmonic mean of the precision and recall[28].

$$\textbf{F1 score} = 2 * \frac{Precision * Recall}{Precision + Recall}$$

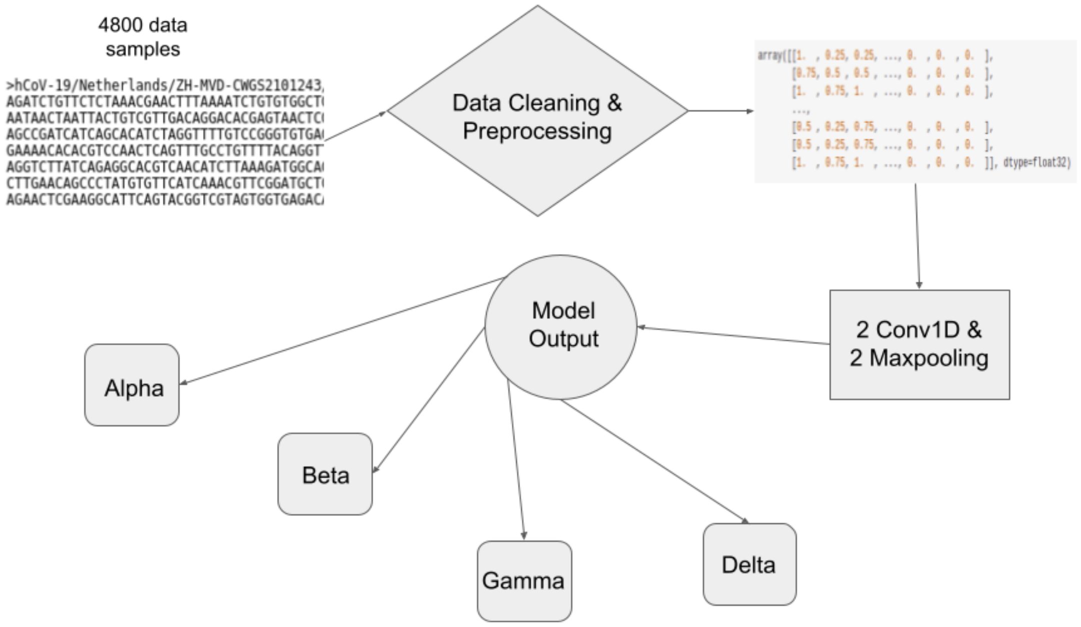
Further, we test our model with more datasets e.g., 4000 data samples, and compare their score with the 2000 data samples, shown in Table 16.

	3200 Data samples		**5200 Data samples**	
Algorithms	**Variant accuracy**	**Variants F1-macro**	**Variants accuracy**	**Variants F1-macro**
Logistic Regression	91.5%	91.5%	91.2%	91.2%

KNN	93.0%	93.0%	93.8%	93.8%
SVM	92.7%	92.7%	93.5%	93.5%
ANN	91.4%	91.0%	88.9%	89.0%
CNN	98.9%	99.0%	96.7%	97.0%

Table 16: Accuracy and F1-macro of all 5 algorithms

From table 14 we see that in both datasets the convolutional neural networks (CNN) accuracy and f1-macro (weighted average of all four variants) are very convincing [21] [29]. We know the nature of CNN, it’s very much famous in machine learning because it identifies unique patterns by itself. So, we do not need to do more things i.e., do more hard coding. The CNN model is shown in fig 1.



```
>hCoV-19/Netherlands/ZH-MVD-CWGS2101243,
AGATCTGTTCTCTAAACGAACTTTAAAATCTGTGTGGCT
AATAACTAATTACTGTCGTTGACAGGACACGAGTAACTC
AGCCGATCATCAGCACATCTAGGTTTTGTCCGGGTGTGA
GAAAACACACGTCCAACTCAGTTTGCCTGTTTTACAGGT
AGGTCTTATCAGAGGCACGTCAACATCTTAAAGATGGCA
CTTGAACAGCCCTATGTGTTCATCAAACGTTCGGATGCT
AGAACTCGAAGGCATTCAGTACGGTCGTAGTGGTGAGAC
```

```
array([[1.  , 0.25, 0.25, ..., 0.  , 0.  , 0.  ],
       [0.75, 0.5 , 0.5 , ..., 0.  , 0.  , 0.  ],
       [1.  , 0.75, 1.  , ..., 0.  , 0.  , 0.  ],
       ...,
       [0.5 , 0.25, 0.75, ..., 0.  , 0.  , 0.  ],
       [0.5 , 0.25, 0.75, ..., 0.  , 0.  , 0.  ],
       [1.  , 0.75, 1.  , ..., 0.  , 0.  , 0.  ]], dtype=float32)
```

Fig 1: CNN Model View

Chapter 4

Conclusion

The world is evolving in all sorts of ways. As well as the technologies. With the aid of artificial intelligence, we can solve critical problems in our life. Artificial intelligence can contribute in various fields e.g., business, marketing, medicine, prediction, and so on. We know that SARS-CoV-2 variants mutate and we can't stop the mutation, it is the nature of the viruses. Therefore, we need to train more datasets or more datasets with different time frames to keep the model act more accurately. This will be our future work where we will take the different datasets from the different time frames. Also, we want to apply more algorithms e.g., LSTM, CNN-BiLSTM GRU which is called deeper learning. [21][29][30][31][32]

References:

[1] "Severe Acute Respiratory Syndrome (SARS)," [Online]. Available: https://www.who.int/health-topics/severe-acute-respiratory-syndrome#tab=tab_1.

[2] "Middle East respiratory syndrome coronavirus (MERS-CoV)," [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/middle-east-respiratory-syndrome-coronavirus-(mers-cov).

[3] "Transmission of SARS-CoV-2: implications for infection," [Online]. Available: https://www.who.int/news-room/commentaries/detail/transmission-of-sars-cov-2-implications-for-infection-prevention-precautions.

[4] "SARS-CoV-2 Variant Classifications and Definitions," [Online]. Available: https://www.cdc.gov/coronavirus/2019-ncov/variants/variant-info.html.

[5] "Variants name," [Online]. Available: https://www.nature.com/articles/d41586-021-01483-0.

[6] "Bangladesh: First cases of COVID-19 confirmed March 8," [Online]. Available: https://www.garda.com/crisis24/news-alerts/320606/bangladesh-first-cases-of-covid-19-confirmed-march-8.

[7] "Five Covid variants found in Bangladesh, so far," [Online]. Available: https://www.dhakatribune.com/health/coronavirus/2021/05/17/five-covid-variants-found-in-bangladesh-so-far.

[8] A. Pavlova *et al.*, "Machine Learning Reveals the Critical Interactions for SARS-CoV-2 Spike Protein Binding to ACE2," *J. Phys. Chem. Lett.*, vol. 12, no. 23, pp. 5494–5502, 2021, doi: 10.1021/acs.jpclett.1c01494.

[9] N. G. Nguyen *et al.*, "DNA Sequence Classification by Convolutional Neural Network," *J. Biomed. Sci. Eng.*, vol. 09, no. 05, pp. 280–286, 2016, doi: 10.4236/jbise.2016.95021.

[10] S. Hu, R. Ma, and H. Wang, "An improved deep learning method for predicting DNA-binding proteins based on contextual features in amino acid sequences," *PLoS One*, vol. 14, no. 11, pp. 1–21, 2019, doi: 10.1371/journal.pone.0225317.

[11] A. Lopez-Rincon *et al.*, "Classification and specific primer design for accurate detection of SARS-CoV-2 using deep learning," *Sci. Rep.*, vol. 11, no. 1, pp. 1–11, 2021, doi: 10.1038/s41598-020-80363-5.

[12] A. Lopez-Rincon *et al.*, "Design of Specific Primer Sets for the Detection of B.1.1.7,

B.1.351 and P.1 SARS-CoV-2 Variants using Deep Learning,” *bioRxiv*, vol. 70, p. 2021.01.20.427043, 2021, [Online]. Available: https://doi.org/10.1101/2021.01.20.427043.

[13] “Human cases of influenza A/H5N8 virus infection,” [Online]. Available: https://www.gisaid.org/.

[14] P. D. Lawrence C. Brody, “Nucleotide,” [Online]. Available: https://www.genome.gov/genetics-glossary/Nucleotide.

[15] A. Singh, “How to handle ‘N’ in Nucleotide/Genes Sequences retrieved from NCBI GeneBank?”

[16] “FASTA Format,” [Online]. Available: https://en.wikipedia.org/wiki/FASTA_format.

[17] “Ordinary least squares Linear Regression.,” [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html.

[18] “Logistic Regression (aka logit, MaxEnt) classifier,” [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html.

[19] “Classifier implementing the k-nearest neighbors,” [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html.

[20] “C-Support Vector Classification.,” [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html.

[21] H. W. Siquan Hu, Ruixiong Ma, “An improved deep learning method for predicting DNA-binding proteins based on contextual features in amino acid sequences,” [Online]. Available: https://www.ncbi.nlm.nih.gov/p0mc/articles/PMC6855455/.

[22] “Short Introduction to Use Keras,” [Online]. Available: https://www.tensorflow.org/tutorials/quickstart/beginner.

[23] “1D convolution layer,” [Online]. Available: https://www.tensorflow.org/api_docs/python/tf/keras/layers/Conv1D.

[24] “A regularizer that applies a L2 regularization penalty.,” [Online]. Available: https://www.tensorflow.org/api_docs/python/tf/keras/regularizers/L2.

[25] “Max pooling Operation for 1D data,” [Online]. Available: https://www.tensorflow.org/api_docs/python/tf/keras/layers/MaxPool1D.

[26] P. E. Romero, “Genetic variants and source of introduction of SARS-CoV-2 in South America,” [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7300916/.

[27] “GISAID Database,” [Online]. Available: https://www.epicov.org/epi3/frontend#4aa0a4.

[28] P. HUILGOL, “Precision vs. Recall – An Intuitive Guide for Every Machine Learning Person,” [Online]. Available: https://www.analyticsvidhya.com/blog/2020/09/precision-recall-machine-learning/.

[29] S. J. & Y. L. Yongqing Zhang, Shaojie Qiao, “DeepSite: bidirectional LSTM and CNN models for predicting DNA–protein binding,” [Online]. Available: https://link.springer.com/article/10.1007/s13042-019-00990-x.

[30] C.-L. X. Qian Liu, Li Fang, Guoliang Yu, Depeng Wang, “Detection of DNA base modifications by deep recurrent neural network on Oxford Nanopore sequencing data,” [Online]. Available: https://www.nature.com/articles/s41467-019-10168-2.

[31] A. S. and E. H. H. Takwa Mohamed, Sabah Sayed, “Long Short-Term Memory Neural Networks for RNA Viruses Mutations Prediction,” [Online]. Available: https://www.hindawi.com/journals/mpe/2021/9980347/.

[32] N. Oskolkov, “LSTM to Detect Neanderthal DNA,” [Online]. Available: https://towardsdatascience.com/lstm-to-detect-neanderthal-dna-843df7e85743.