الجمهورية الجزائرية الديمقراطية الشعبية

People’s Democratic Republic of Algeria

وزارة التعليم العالي والبحث العلمي

Ministry of Higher Education and Scientific Research

Frères Mentouri
Constantine 1 University
Faculty of Natural and Life sciences
Department of Applied Biology

جامعة الإخوة منتوري قسنطينة1
كلية علوم الطبيعة و الحياة
قسم البيولوجيا التطبيقية

Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master in Bioinformatics

SARS-COV 2 Variants and Geo-localization Tracking: A New Deep-learning Approach

Submitted by:

Mohamed el Amine SAYAD

Raid SERRAR

Sustained on: 15 - 07 - 2021

Board of Examiners:

Chairman: Pr. M.A HAMIDECHI

Supervisor: Dr. H. CHEHILI

Examiner: Mr. K. KELLOU

Academic year: 2020-2021

ACKNOWLEDGEMENTS

We express our deep gratitude to our supervisor Dr. Hamza CHEHILI for guiding and allowing us carry out this work.

We also thank Dr. Hamza CHEHILI for his presence, his availability, relevant reflections, guidance, precious advice, patient, and involvement and for his help in writing this dissertation.

Words are insufficient to describe our gratitude to Dr. Hamza CHEHILI. Thanks again.

Our thanks go also to the members of the jury and Prof. Hamidechi A and Mr. Kellou K for their interest in our research by agreeing to evaluate our graduation memory and enrich it by their contribution. We would also like to thank all our teachers who have provided our training throughout these years. We emphasize that their efforts have been beneficial to us in carrying out this work.

We would like that little bit of road that we have achieved together by being master's students give its fruits and that we can elaborate and complete our way through a PhD research project.

We dedicate this work to our parents, for all their sacrifices, support and prayers throughout all university years, to our brothers and friends for their support and constant encouragement.

Abstract

On January 30, 2020, the World Health Organization declared the SARS-CoV-2 epidemic a public health emergency of international concern, detecting emerged variants and tracking them geographically stays a big challenge for the scientific community due to the massive increase in genomic data generated from extensive sequencing of the virus.

This study aims to propose two deep learning models with k-mer preprocessing that can handle and analyze these data, to extract features to identify variants and geographic patterns of their evolution. As a result, the first model exceeded state-of-the-art results while, the second model achieved state-of-the-art results.

Key words: Deep learning; detecting; variants; geographic; preprocessing.

RÉSUMÉ

Le 30 janvier 2020, l'Organisation mondiale de la santé a déclaré l'épidémie de SRAS-CoV-2 comme une urgence de santé publique de portée internationale. La détection des variantes émergentes et leur suivi géographique restent un grand défi pour la communauté scientifique en raison de l'augmentation massive des données génomiques générées par le séquençage extensif du virus.

Cette étude vise à proposer deux modèles d'apprentissage profond avec prétraitement k-mer qui peuvent traiter et analyser ces données, afin d'extraire des caractéristiques pour identifier les variants et les paternes géographiques de leur évolution. En conséquence, le premier modèle a dépassé les résultats de l'état de l'art tandis que le second modèle a atteint les résultats de l'état de l'art.

Mots clés : Apprentissage profond ; détection ; variantes ; géographiques ; prétraitement.

الملخص

فى يوم 30, 2020 يناير أعلنت منظمة الصحة العالمية ان وباء السارس - كوف - 2 هو حالة طوارئ صحية عامة تثير قلقا دوليا ، وان اكتشاف المتغيرات الناشئة وتعقبها جغرافيا يظل تحديا كبيرا للمجتمع العلمى بسبب الزيادة الهائلة في بيانات الجينوم الناتجة عن التسلسل الشامل للفيروس.

تهدف هذه الدراسة إلى اقتراح نموذجين للتعلم العميق مع المعالجة المسبقة التي يمكن أن تتعامل مع هذه البيانات وتحللها، لاستخلاص خصائص لتحديد المتغيرات والأنماط الجغرافية لتطورها. ونتيجة لذلك، تجاوز النموذج الأول النتائج المتوقعة، بينما حقق النموذج الثاني نتائج عالية.

الكلمات المفتاحية : التعلم العميق ، اكتشاف ، المتغيرات ، الجغرافية ، المعالجة.

List of Figures

Figure 1: Phylogenetic relationships of coronaviruses…………………………………….. 3

Figure 2: Schematic representation of the genomic organization of Sars-Cov 2…………… 4

Figure 3: The taxonomy of AI……………………………………………………………10

Figure 4: A structure of an artificial neural network………………………………………...12

Figure 5: The basic structure of a perceptron………………………………………………..13

Figure 6: Example of a multi-fasta file…………………………………………………... 16

Figure 7: Deep learning process applied to achieve the purpose of the approach………….. 19

Figure 8: Data frame used for training the models…………………………………........ 20

Figure 9: A histogram that represent the geographic distribution of Sars-Cov 2 genomes in our dataset.……………………………………………………………………………...20

Figure 10: A histogram that represent the distribution of Sars-Cov 2 genomes per variant in our dataset……………………………………………………………………………….21

Figure 11: Model Summary………………………………………………………………23

Figure 12: Training process……………………………………………………………….24

Figure 13: Model 1 graph over the learning curve of accuracy…………………………….. 26

Figure 14: Model 1 graph over the learning curve of loss………………………………….. 26

Figure 15: Model 1 graph over the learning curve of precision……………………………..26

Figure 16: Model 1 graph over the learning curve of AUC…………………………………26

Figure 17: Model 2 graph over the learning curve of accuracy……………………………27

Figure 18: Model 2 graph over the learning curve of loss…………………………..…... 27

Figure 19: Model 2 graph over the learning curve of precision………………………….... 27

Figure 20: Model 2 graph over the learning curve of AUC………………………………...27

Figure 21: Model 1 graph over the learning curve of ROC …………………….............. 27

Figure 22: Model 2 graph over the learning curve of ROC………………………………… 27

Figure 23: classification report obtained from validation dataset of model 1……………….28

Figure 24: confusion matrix obtained from validation dataset of model 1……………........ 28

Figure 25: confusion matrix obtained from validation dataset of model 2…………………29

Figure 26: classification report obtained from validation dataset of model 2……………….30

List of Tables

Table 1: Structure of Sars-Cov 2 genome……………………………………...………………4

Table 2: Variants of Interest. ……………………………………………………….…….6

Table 3: Variants of concern…………………………………………………………..…7

Table 4: Data used for model training…………………………………………………..…16

Table 5: Basic information on computer configuration that was used for data preparation…18

Table 6: Basic information on High Processing Center (HPC) configuration that was used for data preprocessing and model training……………………………………………….……19

Table 7: Comparison of VarTrack to other related work……………………………………32

ACRONYMS

-CNN: Convolutional Neural Network

-CSV: Comma-Separated Values

-TSV: Tabular-Separated Values

-DL: Deep Learning

-HPC: High-Performance Computing

-IA: Artificial Intelligence

-ML: Machine Learning

-VOC: Variants of concern

-VOI: Variants of interest

-TP: True positive

-TN: True negative

- FP: False positive

-FN: False negative

-TPR: True positive rate

-TNR: True negative rate

-FPR: False positive rate

-FNR: False negative rate

TABLE OF CONTENTS

ACKNOWLEDGEMENTS i

Abstract ii

List of Figures v

List of Tables vi

ACRONYMS vii

Table of Contents viii

Introduction 1

PART ONE: BIBLIOGRAPHIC RESEARCH 3

CHAPTER 1: Insights into SARS-COV 2 Biology 3

1 Origin and diversity 3

2 Genome Organization and structure 3

3 Host receptor usage 5

4 Variants of Sars-Cov 2 5

4.1 Variants of Interest (VOI) 5

4.2 Variant of Concern (VOC) 6

5 Genomic surveillance 7

CHAPTER 2: Deep Learning 9

Introduction 9

1 Artificial intelligence 9

2 Machine Learning 10

3 Supervised learning 11

4 Classification in Machine Learning 11

5 Deep learning 11

5.1 Neural networks 11

5.2 Artificial Neuron (Perceptron) 12

5.3 Convolutional neural networks (CNN) 13

6 Natural language processing: 14

6.1 Classification and Categorization: ... 14

PART TWO: Materials and Methods ... 16

1 Materials ... 16

1.1 Data ... 16

1.2 Software ... 17

1.2.1 Environments ... 17

1.2.2 Packages ... 17

1.3 Hardware ... 18

2 Methods ... 19

2.1 Data preparation ... 19

2.2 Data Visualization ... 20

2.3 Data preprocessing: ... 21

2.4 Model Choosing ... 22

2.5 Make Prediction ... 24

PART THREE: Results and Discussion ... 25

Results ... 25

Discussion ... 32

Conclusion ... 33

References

INTRODUCTION

Introduction

Three coronaviruses have crossed the species barrier to cause deadly pneumonia in humans since the beginning of the 21st century: severe acute respiratory syndrome coronavirus (SARS-CoV), Middle East respiratory syndrome coronavirus (MERS-CoV), and SARS-CoV 2. SARS-CoV emerged within the Guangdong of China in 2002 and deployed to 5 continents through air routes, infecting 8098 folks and inflicting 774 deaths. In 2012, MERS-CoV emerged within the peninsula, wherever it remains a serious public health concern, and was exported to twenty-seven countries, infecting a complete of two,494 people and claiming 858 lives.

SARS-CoV-2 was discovered in December 2019 in Wuhan, Hubei province of China and was sequenced and isolated by January 2020, On January 30, 2020 the World Health Organization declared the SARS-CoV-2 epidemic a public health emergency of international concern, it is still associated with an ongoing outbreak of atypical pneumonia (Covid-19) that has affected over 100 million people and killed more than 4 million of those affected in >60 countries in the past two years [1].

In multiple countries of the world, obtaining new SARS-CoV 2 genomes is ongoing and this will allocate monitoring multiple aspects of this pandemic. These include genetic diversity associated with clinical and epidemiological patterns and profiles, to support detection of new variants and monitor their spread, the differences in speed and scale between the genomic responses during the three pandemics explains the increasing importance of investigating genomic epidemiological data of the virus.

With the increasing number of databases and various curated online repositories, bioinformatics has become a veritable platform for data obtained from genomic epidemiology, the exploitation of bioinformatic tools and techniques and some major in silico studies have led to characterization and structuring of the virus. By aligning these genomic sequences to a reference SARS-CoV-2 genome, numerous mutation sites are identified and the interpretation of phylogenetic relationships between clades/lineages allows to estimate the most probable ancestor and where it originated from by comparison and geographical location [2].

However, some genome annotations are not always stable, given inaccuracies and temporary assignments due to limited information, knowledge, or characterization, in some cases. Also, since there is no taxonomic “ground truth,” taxonomic labels can be subject to dispute, and as methods for determining phylogeny, evolutionary relationships, and taxonomy evolved from physical to molecular characteristics, this sometimes resulted in a series of changes in taxonomic assignments.

On the other hand, the exponential growth of genome-wide assays, and their public access open new horizons for machine learning (ML) methodologies and especially Deep Learning (DL) approaches to effectively perform genetic analysis.

Deep Learning (DL) is a relatively new field compared to traditional techniques, and the application of DL in bioinformatics is an even newer field. However, the last decade has witnessed the rapid development of DL with thrillingly promising power to mine complex relationships hidden in large scale biological and biomedical data, to a new ability to investigate genomic epidemiology in near-real time and to accurately forecast future outbreaks.

The aim is to initiate new approaches for analysis, identifying variants and understanding their geographical distribution in a short period of time and with less computational power, improve accuracy and sustainability of tracking variants.

This thesis is divided into four chapters

- The first chapter discusses SARS-CoV-2 biology (origins, genome structure, and host receptor usage), Variation and classification of variants, monitoring variants from different perspectives.
- In chapter 2 we break down the artificial intelligence concepts and discuss its subfields and components (machine learning, neural networks, deep learning, Natural language processing)
- The third chapter outlines the experimental work carried out in the project.

- Chapter 4 contains a description about the main findings of our research, whereas the discussion section interprets the results that provide the significance of the findings.

PART ONE:

BIBLIOGRAPHIC RESEARCH

CHAPTER 1:

Insights into SARS-COV 2 Biology

1 Origin and diversity

Coronaviruses are RNA viruses globally distributed in an exceeding unknown range of animal species. Coronaviruses vital for humans are found among phylogenetically distinct assortment subgroups, labeled the α- and β-Coronaviruses. SARS-CoV-2 presumably originated in haywire and transmitted to humans through a potential host. supported revealed analysis up to now, pangolins are thought-about the foremost probably intermediate hosts. However, as of now, no definitive host has been found [3].

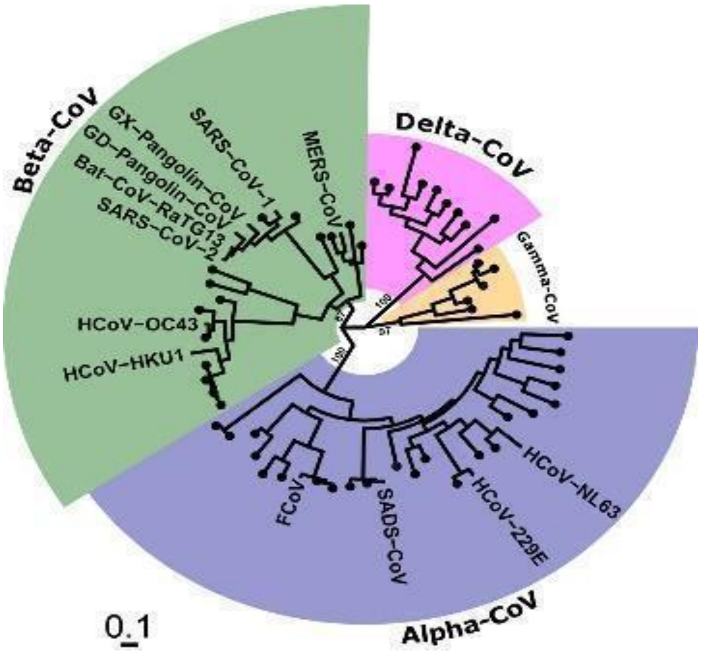


Figure 1: Phylogenetic relationships of coronaviruses [3].

2 Genome Organization and structure

SARS-CoV-2 is a betacoronavirus with swallowed, single-stranded (positive-sense) RNA genomes of animal disease origin. SARS-CoV-2 contains four structural proteins Spike (S), Envelope (E), Membrane (M) and Nucleocapsid (N). The S, M, and E proteins create the envelope of this virus. The E supermolecule, that is that the smallest structural supermolecule, additionally plays a job within the production and maturation of SARS-CoV2 the S and M proteins are concerned within the method of virus attachment throughout replication. The N proteins stay related to the polymer to make a nucleocapsid within the envelope. N is additionally concerned in different aspects of the virus replication cycle (such as assembly and budding) and also the host cellular response to virus infection. This virus is called coronavirus because of the crown-like look of the S glycoprotein [4,5].

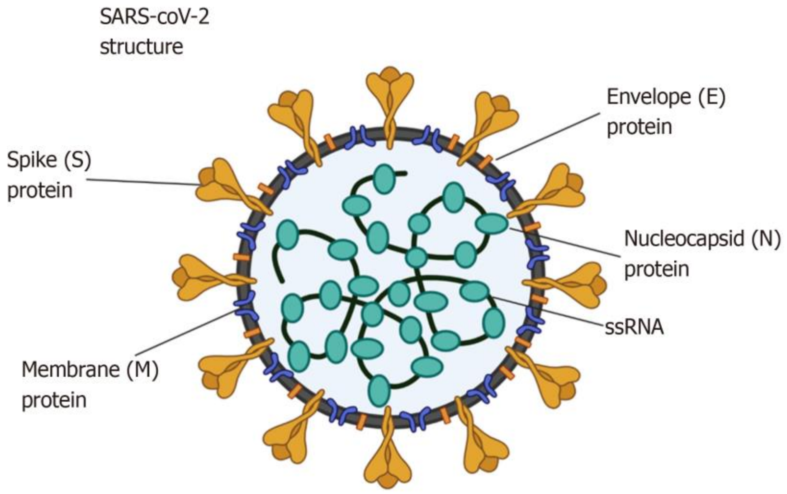


Figure 2: Schematic representation of the genomic organization of Sars-Cov 2 [6].

The genome size of the SARS-CoV-2 varies from 29.8 kb to approximately 30 kb, it is complemented by about six to twelve open reading frames (ORFs) that encodes about 7096 residues long polyprotein which consists of many structural and non-structural proteins (NSPs). The untranslated regions (5'UTR and 3'UTR) are responsible for inter- and intra-molecular interactions, RNA-RNA interactions and for binding the viral and cellular proteins [4].

Table 1: Structure of Sars-Cov 2 genome [5].

5UTR	Orf1ab Gene	S Gene	Orf3a Gene	E Gene	M Gene	Orf6a Gene	Orf7a Gene	Orf7b Gene	Orf8 Gene	N gene Gene	Orf10 Gene	3UTR
Non-coding sequence 265nt	21290nt	3822nt	828nt	228nt	669nt	186nt	366nt	132nt	193nt	908nt	117nt	Non coding sequence 229nt
	Orf1ab Poly-protein	Spike glycol-protein	Orf3a protein	Envelope Protein	Membrane Protein	ORF6 protein	Orf7a Protein	Orf7b Protein	Orf8 Protein	Nucleocapsid Phospho-protein	Orf10 Protein	

3 Host receptor usage

SARS-CoV-2 uses its spike glycoprotein (S) to bind to the same angiotensin-converting enzyme 2 (ACE2) cell receptor as the 2002–2003 SARS-Cov with higher affinity. Most amino acid (AA) residues that are known to be essential for ACE2 binding by SARS-Cov are conserved in SARS-CoV-2 [7].

The total length of S glycoprotein is 1273 AA and consists of a signal peptide (amino acids 1–13) located at the N-terminus and two subunit the S1 subunit (14–685 residues), and the S2 subunit (686–1273 residues); the last two regions are responsible for receptor binding and membrane fusion respectively [8].

4 Variants of Sars-Cov 2

A virus variant is an isolate whose order sequence differs from that of a reference genome, no illation is formed regarding whether or not the modification in order sequence causes any modification within the phenotype of the virus. Viruses perpetually change through mutations and new variants of a virus are expected to occur over time, in most cases the fate of a recently arising mutation is decided by Natural selection. However, mutations may also increase and reduce in frequency because of probability events. For example, a "founder effect" happens once a restricted range of individual viruses establish a brand-new population throughout transmission. The mutations within the genomes of those ancestors can dominate the population despite their effects on viral fitness. This same interaction of Natural selection and likelihood events shapes virus evolution among hosts, in communities, and across countries [9,10].

Variants of SARS-CoV-2 can have completely different characteristics. as an example, some could unfold easily or show signs of resistance to existing treatments and some do not have any impact compared on the present virus, based on spike glycoprotein mutations the CDC unitedly with a SARS-CoV-2 Interagency cluster (SIG) established two classifications for the SARS-CoV-2 variants being monitored: Variant of Interest (VOI), Variant of Concern (VOC) [11].

4.1 Variants of Interest (VOI)

A SARS-CoV-2 isolate may be a VOI if it's phenotypically changed compared to a reference isolate or incorporates a genome with mutations that lead to amino acid changes associated with established or suspected phenotypic implications, and has been noted to cause community transmission/multiple COVID-19 cases/clusters, or has been detected in multiple countries; otherwise assessed to be a VOI by World Health Organization (WHO) in consultation with the UN agency SARS-CoV-2 Virus Evolution unit [10].

Table 2: Variants of Interest [10].

Name	Spike Protein Substitutions	WHO Label	First Identified	Attributes
B.1.525	Spike: A67V, 69del, 70del, 144del, E484K, D614G, Q677H, F888L	Eta	United Kingdom/Nigeria – December 2020	-Potential reduction in neutralization by some antibody treatments and convalescent and post-vaccination sera
B.1.526	Spike: (L5F*), T95I, D253G, (S477N*), (E484K*), D614G, (A701V*)	Iota	United States (New York) – November 2020	Reduced susceptibility to the combination of antibody treatment, Alternative antibody treatments are available Reduced neutralization by convalescent and post-vaccination sera
B.1.617	Spike: L452R, E484Q, D614G	delta	India – February 2021	Potential reduction in neutralization by some EUA monoclonal antibody treatments Reduced neutralization by post-vaccination sera
P.2	Spike: E484K, (F565L*), D614G, V1176F	Zeta	Brazil – April 2020	Potential reduction in neutralization by some EUA monoclonal antibody treatments Reduced neutralization by post-vaccination sera

4.2 Variant of Concern (VOC)

A variant that has been demonstrated to be associated with one or a lot of the subsequent changes at a degree of worldwide public health significance:

- Increase in transmissibility or harmful amendment in COVID-19 epidemiology.
- Increase in virulence or amendment in clinical sickness presentation.
- Decrease in effectiveness of public health and social measures or offered medical specialty, vaccines, therapeutics [10].

Table 3 : Variants of concern[10].

Name	Spike Protein Substitutions	WHO Label	First Identified	Attributes
B.1.1.7	69del, 70del, 144del, (E484K*), (S494P*), N501Y, A570D, D614G, P681H, T716I, S982A, D1118H (K1191N*)	Alpha	United Kingdom	~50% increased transmission. Potential increased severity based on hospitalizations and case fatality rates.
B.1.351	D80A, D215G, 241del, 242del, 243del, K417N, E484K, N501Y, D614G, A701V	Beta	South Africa	~50% increased transmission Reduced neutralization by convalescent and post-vaccination sera.
B.1.427	L452R, D614G	Epsilon	United States-(California)	~20% increased transmission. Reduced neutralization by convalescent and post-vaccination sera.
B.1.429	S13I, W152C, L452R, D614G	Epsilon	United States-(California)	~20% increased transmission. Alternative monoclonal antibody treatments are available. Reduced neutralization by convalescent and post-vaccination sera.

5 Genomic surveillance

Genomic surveillance is in the spotlight as scientists race to track emerging variants of Sars-Cov 2, it refers to the systematic collection and analysis of genomes and epidemiological data from across the globe in order to provide insights about how genomic variations influence health and disease [11].

Extensive efforts of whole genome sequencing (WGS) resulted in a constantly growing data, as of time of writing 6-21-2021 more than two million genome sequences have been published to the public non-profit database GISAID. this genomic data helps in understanding the emerging variants of Sars-Cov 2 as well as the study of infection and investigation of spread across countries, it also has been implemented in developing many applications and tools during the pandemic [12], we mention:

- Pangolin: A web Application assigns global lineages to COVID-19 sequences based on the algorithm that was developed by Áine O'Toole, Verity Hill, JT McCrone, Emily Scher and Andrew Rambaut [13].

- Beast2: BEAST2 is a cross-platform program for Bayesian phylogenetic analysis of molecular sequences. It estimates rooted, time-measured phylogenies using strict or relaxed molecular clock models [14].

- Covidex: Covidex was developed as an open-source alignment-free machine learning subtyping tool. It is a shiny app that allows fast and accurate classification of viral genomes in predefined clusters [15].

Genomic surveillance helps track variants that have impact on therapeutic effectiveness and severity of the disease, it helps also guide the public health action in terms of prevention strategies and understanding the routes of transmission (e.g., enhanced vaccination coverage efforts, mitigate the impact of additional waves of infection, avoid lockdowns, save lives and control the pandemic) [11,12].

Chapter 2:

Deep Learning

Introduction

Genomic data is considered to be one of the big data domains, compared to the major generators of big data (astronomy, YouTube, and Twitter) estimates show that genomics is either on par with the most demanding of the domains in terms of data acquisition, storage, distribution, and analysis. In the past two years genomics has witnessed an explosion in genomic data due to covid-19 pandemic that has put the whole world in aberrant urgent need for developing effective responses, this is the first time we're getting to see an outbreak of a new virus and have the scientific community sharing genomic data that is produced from next-generation-sequencing almost in real time. Said Michael Letko a postdoctoral fellow at rocky Mountain Laboratories [16].

With the wealth of data in-hand, although the traditional methods for analysis are likely to be a powerful means of revealing new biological insights of the virus; however, a number of substantial challenges that currently hamper efforts to harness the power of genomic data but the scientific community must overcome them to pursue this important quest.

In this chapter, we will discuss some important Subsets of artificial intelligence aspects like Machine learning, Deep learning, Natural language processing that rise up and meet the computational challenges and complex ambiguous process that these data poses.

1 Artificial intelligence

Artificial intelligence (AI) is probably the defining technology of the last decade, and also the next. The capability of artificial intelligent approaches to far surpassing human actions in terms of data discovery gained the eye of business and analysis communities all over the world and this field of study witnessed fast progress within the past 20 years.

Artificial Intelligence is a smart machine capable of performing tasks that ordinarily require human intelligence, tasks such as reasoning, problem-solving, planning, optimal decision making, sensory perceptions etc [17].

AI has significantly found its applications in almost every business sector, most common domains are social media (e.g., spam detection, sensitive content masking…), navigation (e.g., GPS: improve operational efficiency, analyze road traffic, and optimize routes.), and healthcare to build sophisticated machines that can detect diseases and identify cancer cells, medical intelligence for the discovery of new drug, many of these artificial intelligence systems vary in terms of techniques and subdivide into machine learning, neural networks, deep learning and some of them are powered by direct instructions[17,18].

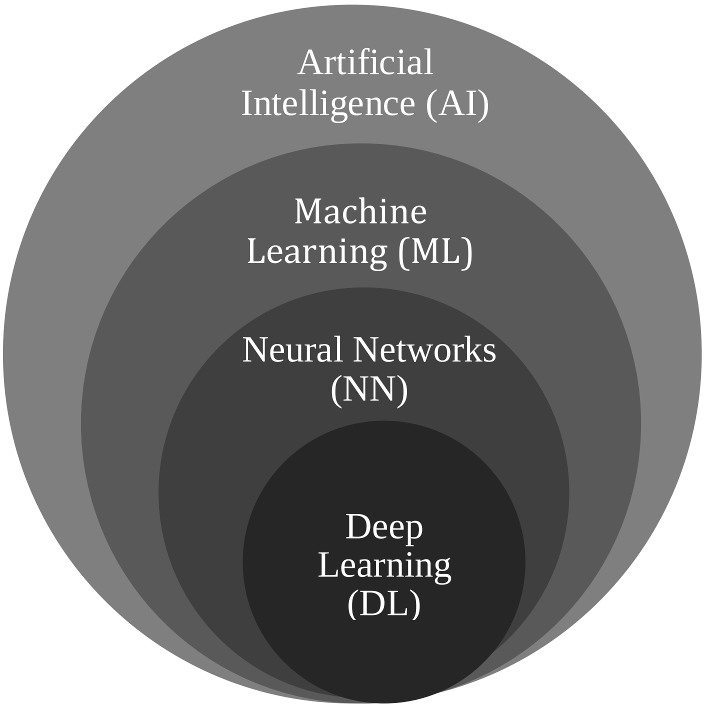


Figure 3: The taxonomy of AI

2 Machine Learning

In 1959, Arthur Samuel, one amongst the pioneers of machine learning, outlined machine learning as a "field of study that offers computers ability to find out without being expressly programmed." Machine Learning is a branch of computer science that can be evaluated from "computational learning theory" in "Artificial intelligence". therefore, rather than you writing the code, what you are doing is feeding information to the generic algorithmic rule, and the algorithm/ machine builds the logic that supported the given data. These programs or algorithms are designed in an exceedingly manner that they learn and improve over time once exposed to new data [19]. Given that the focus of the sector of machine learning is "learning," there are 3 major sorts of learning that are supervised, unsupervised, and reinforcement learning.

in the next section, we will discuss the supervised learning that is associated with our work.

3 Supervised learning

Supervised learning is a task-driven approach, it is the process of inferring a function from huge amounts of data, which has been annotated. Each input of this data is mapped to its corresponding output value “label”, once trained, the algorithm can then apply these labels to new unseen instances using an optimal scenario, by comparing its own output with the correct output to find errors. It then modifies the model accordingly. It can be used to make predictions about unavailable or future data (this is called 'predictive modelling'). The most common supervised tasks are “classification” that separates the data, and “regression” that fits the data [19,20].

4 Classification in Machine Learning

Classification is a technique of categorizing a given set of data into categories, it's performed on every structured or unstructured data. The classification or prognostication modelling is that the task of approximating the mapping operates from input variables to distinct output variables. The goal is to spot that category the new information belongs to [21,22].

5 Deep learning

Deep learning is a subset of machine learning in artificial intelligence-based upon artificial neural networks and representation learning

5.1 Neural networks

Neural networks are a group of algorithms, sculpturesque loosely once the human brain, that is designed to recognize patterns. They interpret sensory knowledge through a kind of machine perception, labeling, or cluster raw input. The patterns they perceive are numerical, contained in vectors, into which all real-world information, be it pictures, sound, text, or statistic, must be translated [23,24].

Artificial neural networks are composed of layers of node

- Each node is designed to behave similarly to a neuron in the brain
- The first layer of a neural net is called the input layer, followed by hidden layers, then finally the output layer
- Each node in the neural net performs some sort of calculation, which is passed on to other nodes deeper in the neural net

Deep is a technical term. It refers to the number of layers in a neural network, depending on the complexity of the problem, the number of layers varies from one to hundreds. A shallow network has one hidden layer, and a deep network has many. Multiple hidden layers permit deep neural networks to extract knowledge from the data in a feature hierarchy [25,26].

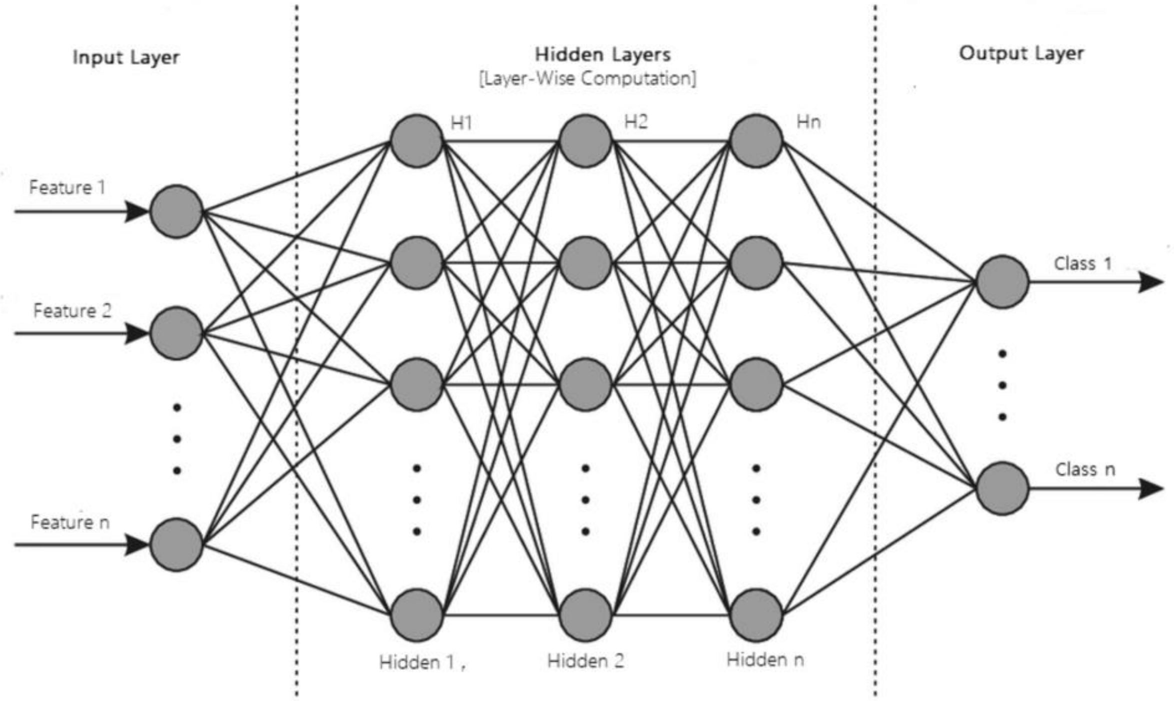


Figure 4: A structure of an artificial neural network modeling with multiple processing layers [27].

In other words, Deep learning removes the manual identification of features in data and, instead, depends on any training method it has to find the important patterns within the input examples, this makes training the neural network easier and quicker, and it can yield a higher result that advances the field of artificial intelligence. These Deep learning models would improve well when more data is added to the architecture. The only difference in the deep learning model is that with experience model becomes better without any specific guidance [28].

5.2 Artificial Neuron (Perceptron)

Artificial neuron conjointly referred to as perceptron is the basic unit of the neural network. In easy terms, it's a function based on a model of biological neurons. it's used for supervised learning of binary classifiers [29].

The perceptron (artificial neuron) consists of four parts:

- Input values or One input layer: input values are passed to a neuron using this layer. it would be one thing as easy as a group of array values. It's just like a dendrite in biological neurons.
- Weights and Bias: Weights are a group of array values that are increased by the various input values. we tend to then take a sum of these multiplied values which is named a weighted

sum. Next, we tend to add a bias value to the weighted sum to induce the ultimate value for prediction by our neuron.

- Activation Function: Activation Function decides whether or not a neuron is fired. It decides which of the two output values should be generated by the neuron.
- Output Layer: The output layer gives the final output of a neuron which can then be passed to other neurons in the network or taken as the final output value [29,30].

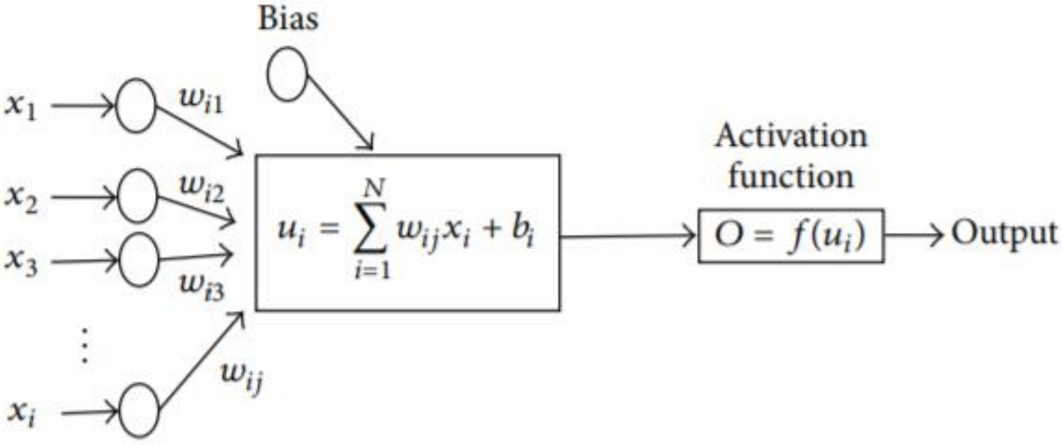


Figure 5: The basic structure of a perceptron [31].

5.3 Convolutional neural networks (CNN)

CNN is a class of deep, feed-forward artificial neural networks (where connections between nodes do not form a cycle) & use a variation of multilayer perceptron designed to require minimal preprocessing. These are inspired by the animal visual cortex [31,32] (e.g: LeNet-5)

LeNet-5 is a convolutional neural network proposed by Yann LeCun in 1989. It was one of the earliest ConvNet architecture and has had a dominant influence over the coming architectures [33].

CNN architectures are basically made of 3 elements:

- Convolution: The term convolution refers to the mathematical combination of two functions to produce a third function.
- Pooling: The objective of Pooling is to down-sample an input representation (image, hidden-layer output matrix, etc.), reducing its dimensions and allowing for assumptions to be made about features contained in the sub-regions created.
- Fully Connected Layers: in a neural network are those layers where all the inputs from one layer are connected to every activation unit of the next layer [39].

6 Natural language processing:

Natural language processing helps process and extract substantive insights from the human language's textual or spoken information, with the assistance of linguistics (rule-based modelling of human language), statistics, machine learning, and deep learning. Specifically, it's a subfield of AI that programs computer to process and analyze massive amounts of linguistic communication information [34].

The Relationship Between NLP and Deep Learning:

Deep learning changes that. In this field, deep learning enables AI to learn the meaning of words or phrases through directly observing how they are used in a paragraph As a result, rather than requiring pre-identified contexts or humans to clarify meanings and outline relations, the which means and relations of words or phrases are learned merely from raw paragraphs. This implies that once an individual searches for 'clear plaster', the AI understands they're more probably to mean the first aid item than the furnishing material, which brings the NLP level nearer to a person's learner's level [34,35].

The major applications of NLP which become easier to solve with deep learning are Text Classification and Categorization, Named Entity Recognition (NER), Part of Speech Tagging Machine Translation, Speech Recognition, Question Answering, Document Summarization [36].

6.1 Classification and Categorization:

Text classification is the method that assigns categories (or labels) to textual data that range from short phrases to much longer documents. sometimes stated as "text categorization" [37,38].

It is a very essential part nowadays, to make many applications such as web searching, email spam filtering, language identification, etc. Currently, most of the companies are also working on product classification, when they are scraping data from different websites and lastly making a taxonomy of map data of different sites and providing automatic product classification.

We can mention some examples:

- NLP enables the recognition and prediction of diseases based on electronic health records and patient's speech. This capability is being explored in health conditions that go from cardiovascular diseases to depression and even schizophrenia. For example, Amazon Comprehend Medical is a service that uses NLP to extract disease conditions, medications, and treatment outcomes from patient notes, clinical trial reports, and other electronic health records [39].

- Organizations can determine what customers are saying about a service or product by identifying and extracting information from sources like social media. This sentiment analysis can provide a lot of information about customers' choices and their decision drivers [40].
- Companies like Google filter and classify your emails with NLP by analyzing text in emails that flow through their servers and stopping spam before they even enter your inbox [41].
- To help to identify fake news, the NLP Group at MIT developed a new system to determine if a source is accurate or politically biased, detecting if a news source can be trusted or not [42].

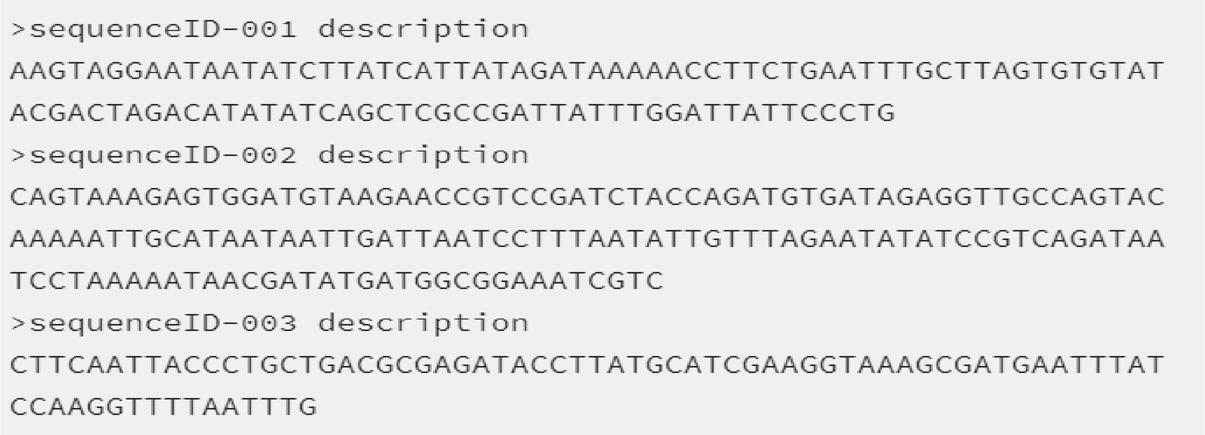
PART TWO:
MATERIELS AND METHODS

1 Materials

1.1 Data

Consists of 2 files of 2 different formats, a multi-FASTA file contains multiple genome sequences of Sars-Cov 2 and a TSV (A tab-separated values) file that contains description of all the records in FASTA file, each record in the table is on a separate line, and data columns represents the field of information, both files were downloaded from GISAID database.

The Fasta format is a text-based format for representing either nucleotide sequences or peptide sequences. A Fasta file begins with a description line which starts with ">" and includes the sequence identifier and a description. The following lines contain the sequence data which are expected to be represented in the standard IUPAC amino acid and nucleic acid codes (Figure6), A tab-separated values (TSV) file is commonly used by spreadsheet applications to exchange data between databases. It stores a data table in which each record in the table is on a separate line, and data columns are separated by tabs.



```
>sequenceID-001 description
AAGTAGGAATAATATCTTATCATTATAGATAAAAACCTTCTGAATTTGCTTAGTGTGTAT
ACGACTAGACATATATCAGCTCGCCGATTATTTGGATTATTCCCTG
>sequenceID-002 description
CAGTAAAGAGTGGATGTAAGAACCGTCCGATCTACCAGATGTGATAGAGGTTGCCAGTAC
AAAAATTGCATAATAATTGATTAATCCTTTAATATTGTTTAGAATATATCCGTCAGATAA
TCCTAAAAATAACGATATGATGGCGGAAATCGTC
>sequenceID-003 description
CTTCAATTACCCTGCTGACGCGAGATACCTTATGCATCGAAGGTAAAGCGATGAATTTAT
CCAAGGTTTTAATTTG
```

Figure 6: Example of a multi-fasta file

A comma-separated values (CSV) file is a delimited text file that uses a comma to separate values, A CSV file typically stores tabular data (numbers and text) in plain text.

Table 4: Data used for model training

Format	Description	Size
FASTA (.fasta)	sequences.fasta	4GB
Tab-separate values file (.tsv)	metadata.tsv	70MB

1.2 Software

1.2.1 Environments

Python

Python is free and simple to learn high-level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation, Python's elegant syntax and dynamic typing, together with its interpreted nature, allows programmers to express concepts in fewer lines of code and make debugging of errors easy.

It is an ideal language for scripting and rapid application development in many areas on most platforms: web development, machine Learning and artificial Intelligence, data Science and data visualization

Anaconda

Anaconda® is a package manager, an environment manager, a Python/R data science distribution, and a collection of over 7,500+ open-source packages. Anaconda is free and easy to install, and it offers free community support.

Jupyter Notebook

The Jupyter Notebook is an open-source web application that allows creating and sharing documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

Visual studio code:

Visual Studio Code is a free coding editor, it has support for many languages, including Python, Java, C++, JavaScript, and more.

1.2.2 Packages

Biopython

Biopython is a set of freely available tools for biological computation written in Python by an international team of developers.

NumPy

NumPy stands for Numerical Python, NumPy was created in 2005 by Travis Oliphant.it is an open free source project that is used for working with arrays.

It also has functions for working in domain of linear algebra, fourier transform, and matrices.

Pandas

Pandas is a fast, powerful, flexible and easy to use open-source data analysis and manipulation tool, it is built on top of the Python programming language.

Matplotlib

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations and plotting in Python. It is used along with NumPy to provide an environment that is an effective open-source alternative for MATLAB. It can also be used with graphics toolkits like PyQt and wxPython.

TensorFlow

TensorFlow is an end-to-end open-source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries, and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML-powered applications.

Sickit-learn

Scikit-learn (also known as sklearn) is a free software machine learning library for Python. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN. It is built upon NumPy, SciPy and Matplotlib

Keras

Keras is a high-level neural network API, written in Python and interfaced with TensorFlow, CNTK and Theano. It has been developed with the objective of enabling rapid experimentation.

Seaborn

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

1.3 Hardware

Table 5: Basic information on computer configuration that was used for data preparation

Computer	Characteristics
Processor	Intel i7-6700HQ CPU @ 2.60GHz
RAM	8 GB DDR4
Storage	256 GB SSD
System	Windows 10 professional

Table 6: Basic information on High Processing Center (HPC) configuration that was used for data preprocessing and model training

High Processing Center	Characteristics
CPU	12 nodes (2*14 cores),
GPU	2 nodes (4 GPU Tesla V100)
RAM	128 GB per node

2 Methods

in this section we arrange a clear and precise description of the chosen experimental procedures and different protocols used in the steps of the deep learning process applied to achieve the purpose of this document in chronological order (Figure 7).

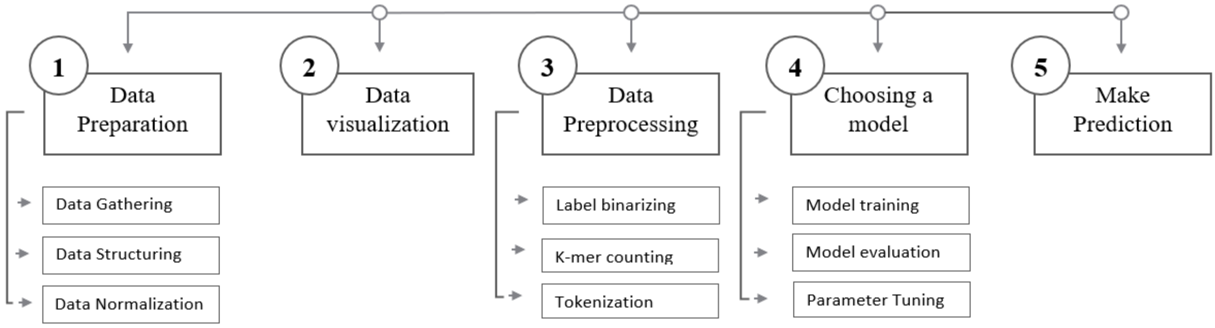


Figure 7: Deep learning process applied to achieve the purpose of the approach.

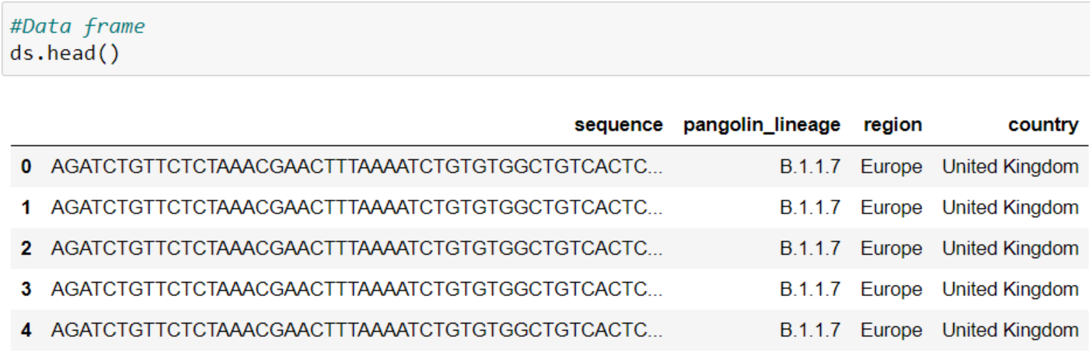
2.1 Data preparation

This section describes the essential steps to construct the dataset that was used for training the model.

One hundred seventy-five thousand nine hundred seventy-six SARS-CoV-2 genome sequences were downloaded from GISAID in chunks of 5000 sequence at a time, only high coverage, human host, and complete sequences were selected, the downloaded file contains a fasta file ‘sequences.fasta’ along with its metadata information ‘metadata.tsv’.

The metadata files were read as a panda’s data-frame in python, and the fasta files were parsed using biopython module ‘SeqIO’, then we assigned every genome to its metadata (e.g.: sample id, geographic location, collection date, originating lab...). all data-frames were then concatenated into one that contains 175976 rows (each corresponding to a variant genome) and 29 columns (each represent a field of information for each variant) (Figure 8).

we sampled the attributes by selecting columns of interest and dropped countries that occur less than 3000 times and only the nine major variant lineages were kept. the dataset was saved in two csv files that are essential for training the model, each for a corresponding target (variant lineage, geographic location).



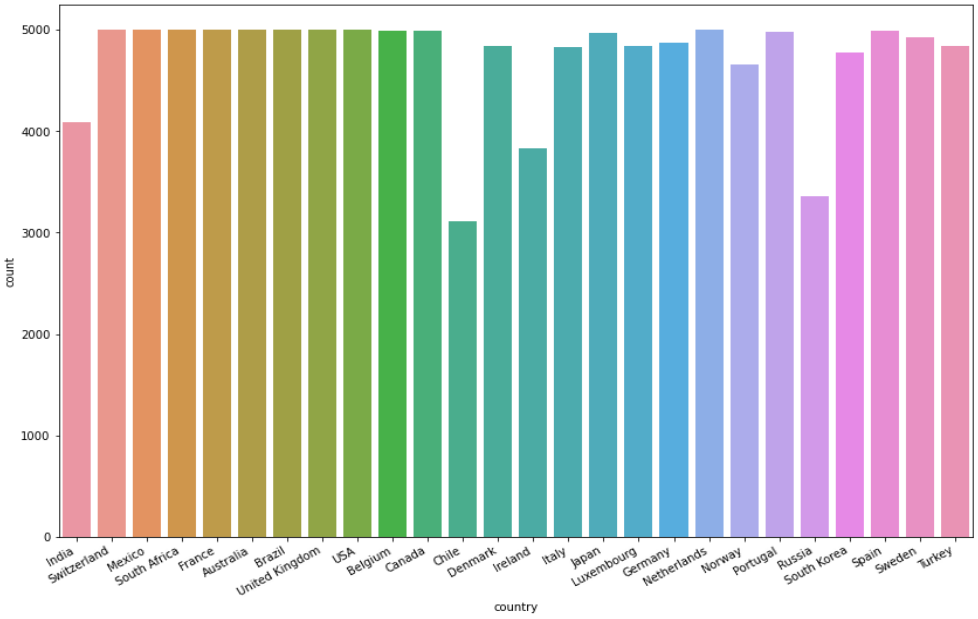
```
#Data frame
ds.head()
```

	sequence	pangolin_lineage	region	country
0	AGATCTGTTCTCTAAACGAACTTTAAAATCTGTGTGGCTGTCACTC...	B.1.1.7	Europe	United Kingdom
1	AGATCTGTTCTCTAAACGAACTTTAAAATCTGTGTGGCTGTCACTC...	B.1.1.7	Europe	United Kingdom
2	AGATCTGTTCTCTAAACGAACTTTAAAATCTGTGTGGCTGTCACTC...	B.1.1.7	Europe	United Kingdom
3	AGATCTGTTCTCTAAACGAACTTTAAAATCTGTGTGGCTGTCACTC...	B.1.1.7	Europe	United Kingdom
4	AGATCTGTTCTCTAAACGAACTTTAAAATCTGTGTGGCTGTCACTC...	B.1.1.7	Europe	United Kingdom

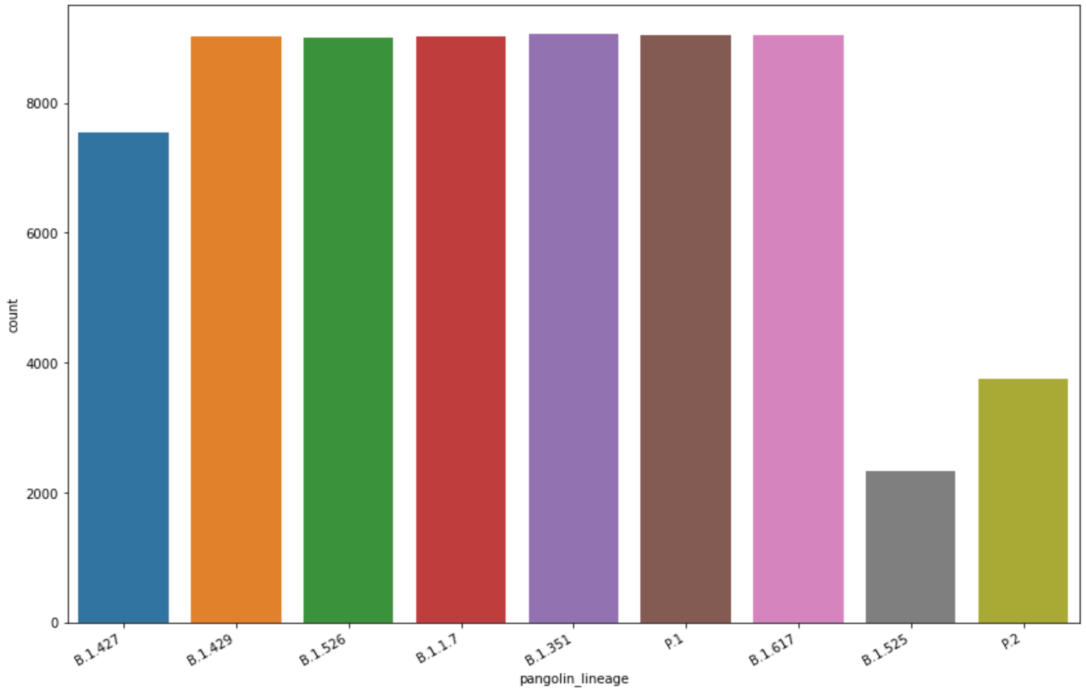
Figure 8: Data frame that used for training the models.

2.2 Data Visualization

we used visual and quantitative methods to understand and summarize the dataset as it is a crucial step before diving into the following steps that include; Bivariate visualization and summary statistics for assessing the relationship between each Variant in the dataset and the target variable of interest (e.g., pangolin lineage, geographic location), Which provided the context needed to develop an appropriate model for the problem at hand and to correctly interpret its results.



<u>Figure 9:</u> A histogram that represent the geographic distribution of Sars-Cov 2 genome sequences in our dataset.



<u>Figure 10:</u> A histogram that represent the distribution of Sars-Cov 2 genome sequences per variant in our dataset.

2.3 Data preprocessing

First, we defined input and output variables of the dataset, then we used Label Binarizer which is a Scikit-Learn class to convert Categorical classes to an NumPy array.

Then we split the dataset into training (75%) and testing (25%) using the scikit-learn ‘train_test_split’ function, so we can easily evaluate the performance of the model on an unseen test dataset.

we applied k-mers on all inputs to break all the genome sequences down into k-mer length overlapping “words”. we used “words” of length 3, “ATGCATGCA” becomes: ‘ATG’, ‘TGC’, ‘GCA’, ‘CAT’,’ATG’,’TGC’,’GCA’.

In genomics, we refer to these types of manipulations as “k-mer counting”, or counting the occurrences of each possible k-mer sequence

For a sequence, if the sequence length is m, then the number of k-mer subsequence of length k has m-k + 1. The sequence generally consists of four bases, A, T, C, and G, and thus k-mers of length k have 4^k possible structures.

We proceed with text preprocessing using the Keras Tokenizer class, this class allows to vectorize a text corpus, by turning each text into a vector where the coefficient for each token is binary based on Word Count, the number of words is the maximum number of words to keep based on word frequency, these sequences are then split into lists of tokens and vectorized.

We then create the word index dictionary based on word frequency with the help of "fit on text" function that updates internal vocabulary based on a list of texts (e.g., the dictionary of the following sequence [agt, ccc, agt, ggt, ccc, agt] will be presented as: ["agt"] = 1; ["ccc"] = 2; ["ggt"] =3, as every word gets a unique integer value starting from one while zero is reserved for padding.

We then transform each text in texts to a sequence of integers using the "texts to sequences" that takes each word in the text and replaces it with its corresponding integer value from the word index dictionary.

At last, we defined the max length of the input to 32000 and padded all the sequences since every sequence in texts does not have the same number of words and the neural network requires it to have inputs of fixed length.

2.4 Model Choosing

we built a variant and geographic location classification model. In Keras, a model is created using Sequential Keras API.

The model created has fully connected layers, which means all the neurons are connected from one layer to its next layer. This is achieved in Keras with the help of the Dense function.

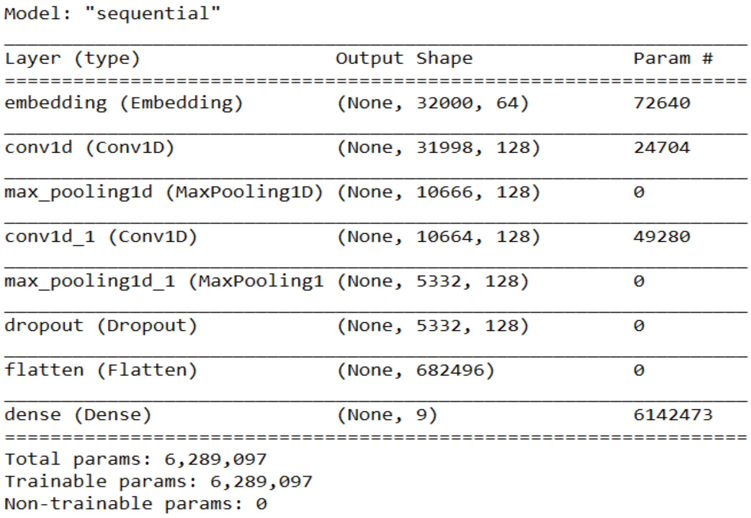
The first layer in our model is a trainable embedding layer. Embedding layers are used to transform discrete inputs to points in vector space, called embedding vectors. Embedding vectors are a staple of natural language processing where they are used to represent words in an L-dimensional space, where L is the length of the vector.

In natural language preprocessing, they represent the relation among input tokens, which are in general words, in a fixed set of possible words

For our model we consider each genome to be a discrete input token, and the set of 64 codons (genetic code) to be our dictionary.

The next layer in our model is a trainable convolutional layer. It receives uniform matrices generated from our embedding layers and applies convolutions using 128 trainable filters, each with a window of size 3 and selecting the Relu activation function.

To reduce overfitting and capturing noise, these feature maps are then sub-sampled using max pooling, with a window size of 3 and a partial neuron connection with a probability of 0.25 are omitted before the fully connected layer. then We flatten the feature maps to 1x1 dimensional matrices, where each matrix represents a feature of the input sequence finally the last layer is the fully connected layer (dense layer) where the number of neurons represents the number of target classes which is 26 countries for geographical location classifier, and 9 lineages for. The extracted features are then used to train our classification model.

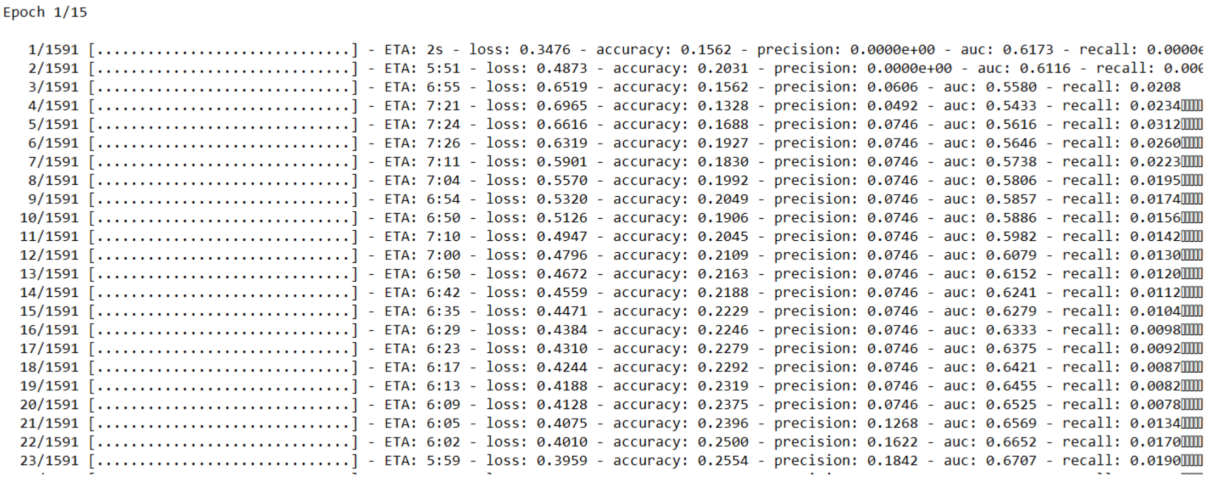


```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, 32000, 64)         72640
_________________________________________________________________
conv1d (Conv1D)              (None, 31998, 128)        24704
_________________________________________________________________
max_pooling1d (MaxPooling1D) (None, 10666, 128)        0
_________________________________________________________________
conv1d_1 (Conv1D)            (None, 10664, 128)        49280
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 5332, 128)         0
_________________________________________________________________
dropout (Dropout)            (None, 5332, 128)         0
_________________________________________________________________
flatten (Flatten)            (None, 682496)            0
_________________________________________________________________
dense (Dense)                (None, 9)                 6142473
=================================================================
Total params: 6,289,097
Trainable params: 6,289,097
Non-trainable params: 0
```

Figure 11: Model Summary.

Training the model

The training process will run with the fit() function for a fixed number of iterations through the dataset called epochs, which we specified using the epochs argument as 15 and We set the number of dataset rows that are considered before the model weights are updated within each epoch to 32 using the batch size argument then we saved the Model weights and architecture after the training process ended as ‘h5’ file. An H5 file is a data file saved in HDF (Hierarchical Data Format), This is a grid format that is ideal for storing multi-dimensional arrays of numbers.



```
Epoch 1/15

   1/1591 [..............................] - ETA: 2s - loss: 0.3476 - accuracy: 0.1562 - precision: 0.0000e+00 - auc: 0.6173 - recall: 0.0000e
   2/1591 [..............................] - ETA: 5:51 - loss: 0.4873 - accuracy: 0.2031 - precision: 0.0000e+00 - auc: 0.6116 - recall: 0.000
   3/1591 [..............................] - ETA: 6:55 - loss: 0.6519 - accuracy: 0.1562 - precision: 0.0606 - auc: 0.5580 - recall: 0.0208
   4/1591 [..............................] - ETA: 7:21 - loss: 0.6965 - accuracy: 0.1328 - precision: 0.0492 - auc: 0.5433 - recall: 0.0234
   5/1591 [..............................] - ETA: 7:24 - loss: 0.6616 - accuracy: 0.1688 - precision: 0.0746 - auc: 0.5616 - recall: 0.0312
   6/1591 [..............................] - ETA: 7:26 - loss: 0.6319 - accuracy: 0.1927 - precision: 0.0746 - auc: 0.5646 - recall: 0.0260
   7/1591 [..............................] - ETA: 7:11 - loss: 0.5901 - accuracy: 0.1830 - precision: 0.0746 - auc: 0.5738 - recall: 0.0223
   8/1591 [..............................] - ETA: 7:04 - loss: 0.5570 - accuracy: 0.1992 - precision: 0.0746 - auc: 0.5806 - recall: 0.0195
   9/1591 [..............................] - ETA: 6:54 - loss: 0.5320 - accuracy: 0.2049 - precision: 0.0746 - auc: 0.5857 - recall: 0.0174
  10/1591 [..............................] - ETA: 6:50 - loss: 0.5126 - accuracy: 0.1906 - precision: 0.0746 - auc: 0.5886 - recall: 0.0156
  11/1591 [..............................] - ETA: 7:10 - loss: 0.4947 - accuracy: 0.2045 - precision: 0.0746 - auc: 0.5982 - recall: 0.0142
  12/1591 [..............................] - ETA: 7:00 - loss: 0.4796 - accuracy: 0.2109 - precision: 0.0746 - auc: 0.6079 - recall: 0.0130
  13/1591 [..............................] - ETA: 6:50 - loss: 0.4672 - accuracy: 0.2163 - precision: 0.0746 - auc: 0.6152 - recall: 0.0120
  14/1591 [..............................] - ETA: 6:42 - loss: 0.4559 - accuracy: 0.2188 - precision: 0.0746 - auc: 0.6241 - recall: 0.0112
  15/1591 [..............................] - ETA: 6:35 - loss: 0.4471 - accuracy: 0.2229 - precision: 0.0746 - auc: 0.6279 - recall: 0.0104
  16/1591 [..............................] - ETA: 6:29 - loss: 0.4384 - accuracy: 0.2246 - precision: 0.0746 - auc: 0.6333 - recall: 0.0098
  17/1591 [..............................] - ETA: 6:23 - loss: 0.4310 - accuracy: 0.2279 - precision: 0.0746 - auc: 0.6375 - recall: 0.0092
  18/1591 [..............................] - ETA: 6:17 - loss: 0.4244 - accuracy: 0.2292 - precision: 0.0746 - auc: 0.6421 - recall: 0.0087
  19/1591 [..............................] - ETA: 6:13 - loss: 0.4188 - accuracy: 0.2319 - precision: 0.0746 - auc: 0.6455 - recall: 0.0082
  20/1591 [..............................] - ETA: 6:09 - loss: 0.4128 - accuracy: 0.2375 - precision: 0.0746 - auc: 0.6525 - recall: 0.0078
  21/1591 [..............................] - ETA: 6:05 - loss: 0.4075 - accuracy: 0.2396 - precision: 0.1268 - auc: 0.6569 - recall: 0.0134
  22/1591 [..............................] - ETA: 6:02 - loss: 0.4010 - accuracy: 0.2500 - precision: 0.1622 - auc: 0.6652 - recall: 0.0170
  23/1591 [..............................] - ETA: 5:59 - loss: 0.3959 - accuracy: 0.2554 - precision: 0.1842 - auc: 0.6707 - recall: 0.0190
```

Figure 12: Training process.

Evaluating the model

To measure the performance of our classification model we load the trained model and evaluate it with an unseen validation dataset using the ‘evaluate ()’ function in Keras. There are various ways to evaluate a deep learning model’s performance, we used a couple of widely used metrics (accuracy, precision, AUC/ROC curve, Recall), and to facilitate the interpretation of our results we used the confusion matrix and classification report.

2.5 Make Prediction:

The prediction stage is the state of the model being put into use, the algorithm will generate probable values for an unknown variable for each record (genome) in the new data which must be run through the same data preprocessing steps (k-mer counting, tokenization) as training stage, allowing the model builder to identify what the value most likely be, which means the model will present us with a direct answer to our problematic and digs out variants, this is achieved using the TensorFlow function (‘model.predict’).

The results of the previous methods are presented in the next chapter (results and discussion).

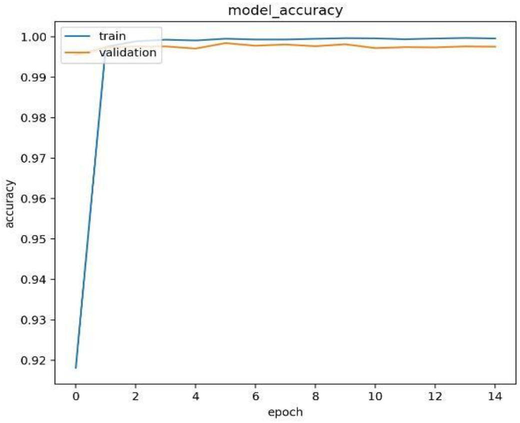
PART THREE:

Results and Discussion

Results

After running our experiments the accuracy achieved on training samples was 99.96% and 99,76% on test samples for the variant classifier, while the accuracy achieved for training and test samples in the geographical classification model are 93,39% and 87,05% respectively can be seen in (Figure 13,17). As shown, we obtained a very good accuracy in all experiments. Besides the accuracy of the model, we also monitored the loss in the training and testing datasets during the learning process, as shown in (Figure 14,18), the precision obtained on train and test samples was 99,96% and 99,76% respectively for the first model while 95,69% and 99,76% approximately for the second model (Figure 15, 19) the AUC obtained on train and test samples was 99,99 % and 99,91% respectively for the first model, while 99,78% and 97,99% approximately for the second model (Figure 16,20). Classifying different viral genomes of sars_cov2 needs accurate preprocessing and careful feature learning due to the high similarity of sequences. As we described in previous chapters, we used very careful preprocessing on our dataset in order to prevent any problem during the analysis. The k-mers preprocessing data has a major influence on the high and consistent accuracy.

After loading the trained model, an evaluation was done on new validation data that was collected separately from training data. Similarly, we obtained a high accuracy of 99,91% with 0,09% error rate for variant classifier and 93,06% with 6,94 % error rate of geographic location classification model, as mentioned in previous chapter the evaluation was done using ROC curve metrics, confusion matrix, and classification report see (Figures 21 ,23 ,24) and for geographical classification model (Figure 22,25,26).



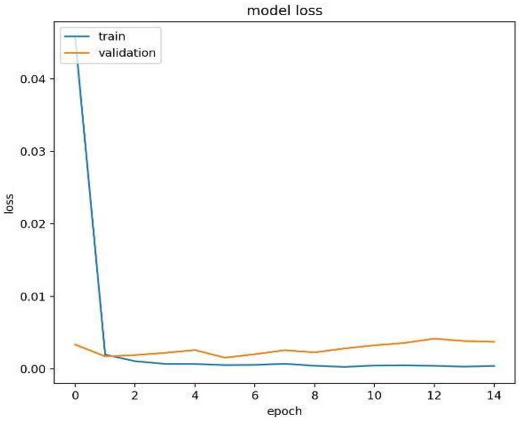
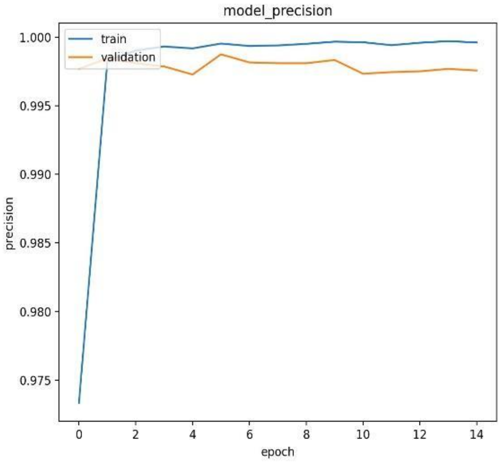


Figure 13: Model 1 graph over the learning curve of accuracy

Figure 14: Model 1 graph over the learning curve of loss



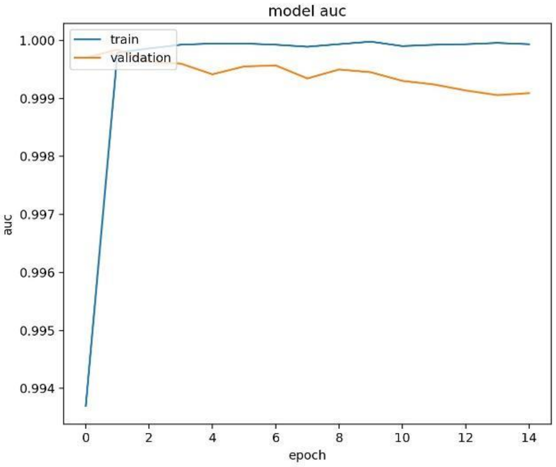
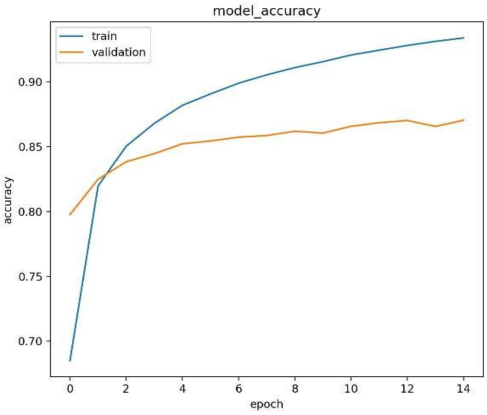


Figure 15: Model 1 graph over the learning curve of precision.

Figure 16: Model 1 graph over the learning curve of AUC



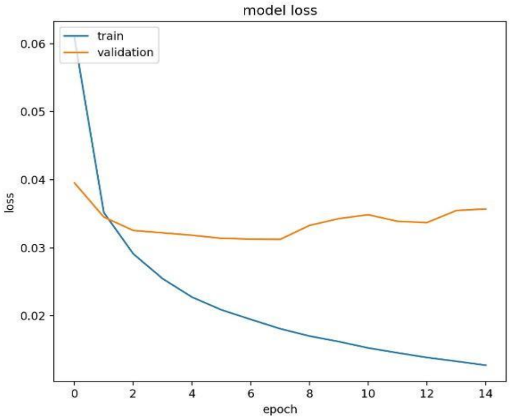
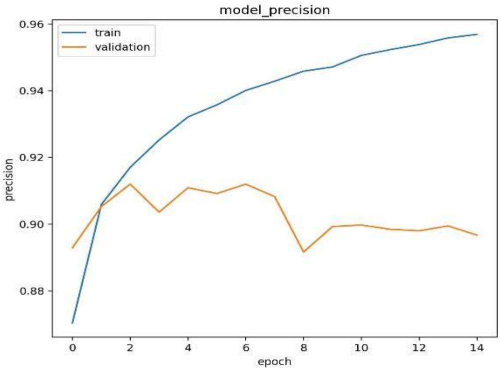


Figure 17: Model 2 graph over the learning curve of accuracy

Figure 18: Model 2 graph over the learning curve of loss



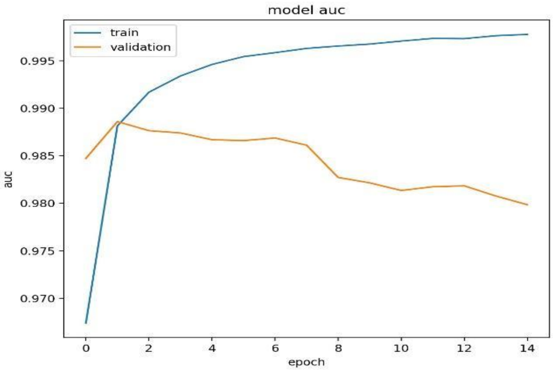
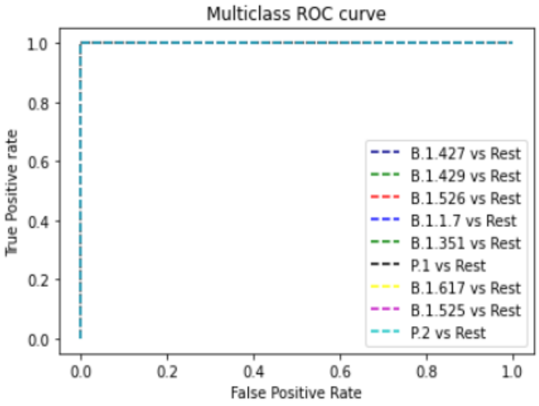


Figure 19: Model 2 graph over the curve of precision

Figure 20: Model 2 graph over the learning curve of AUC



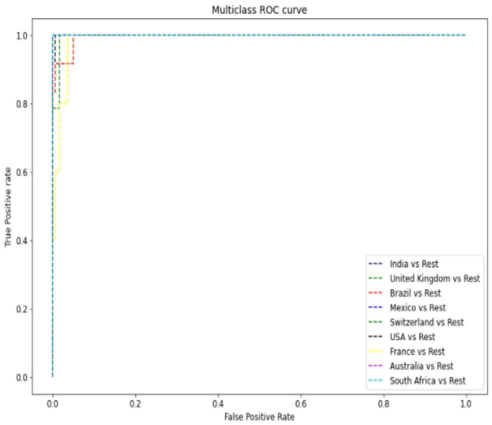


Figure 21: Model 1 graph over the learning curve of ROC

Figure 22: Model 2 graph over the learning curve of ROC

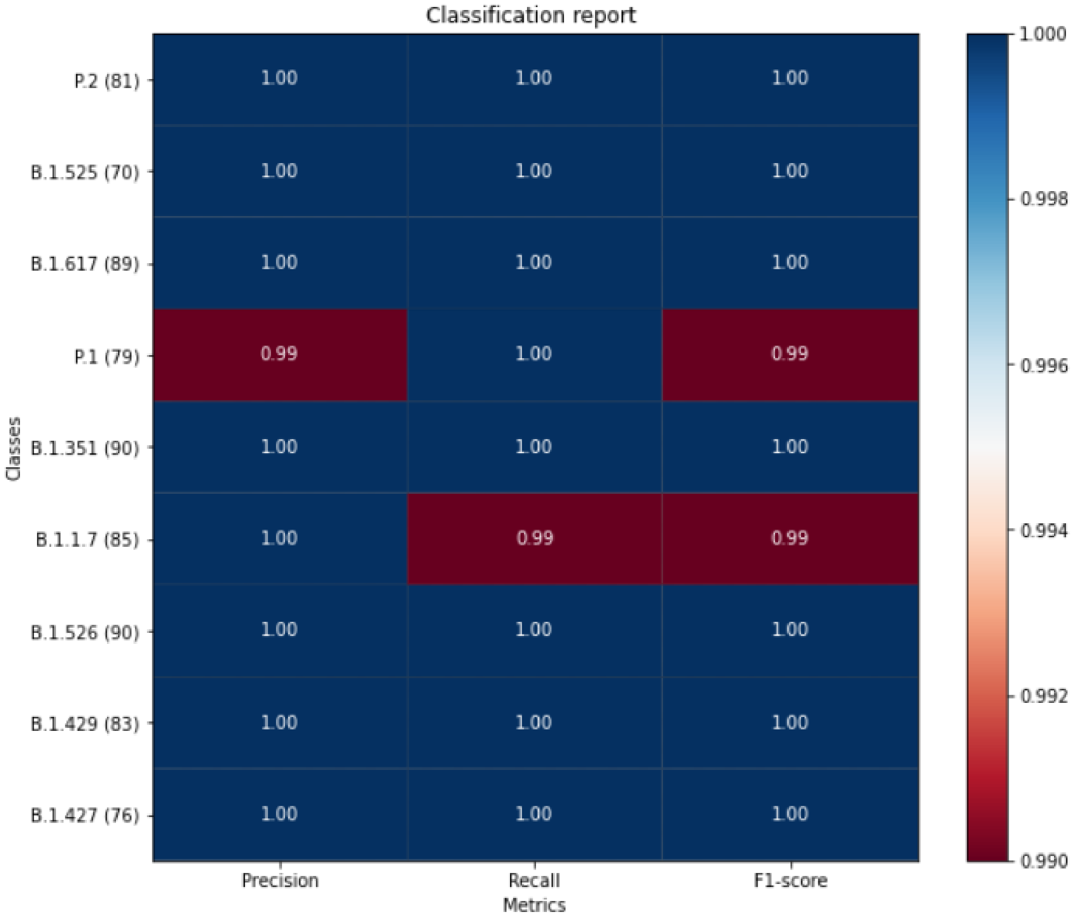


Figure 23: Classification report obtained from validation dataset of model 1

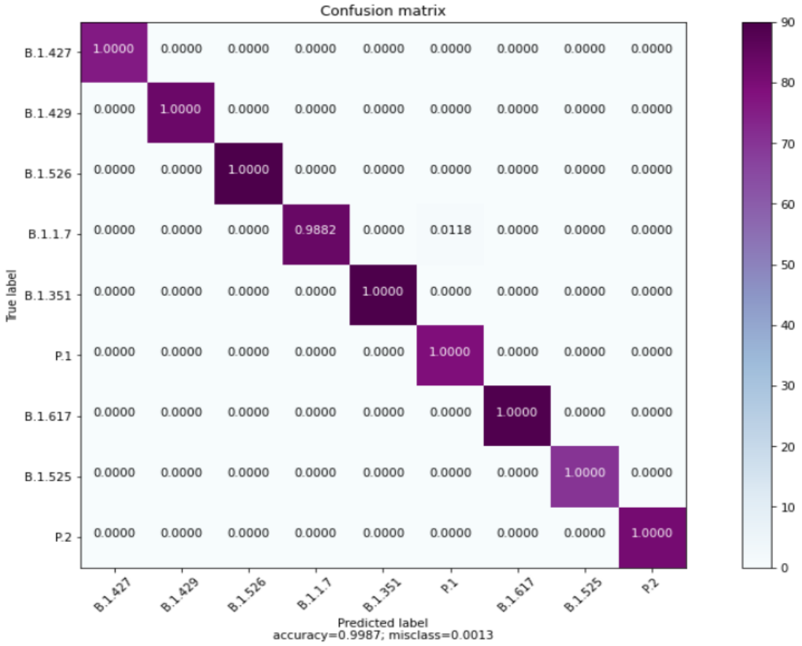


Figure 24: Confusion matrix obtained from validation dataset of model 1.

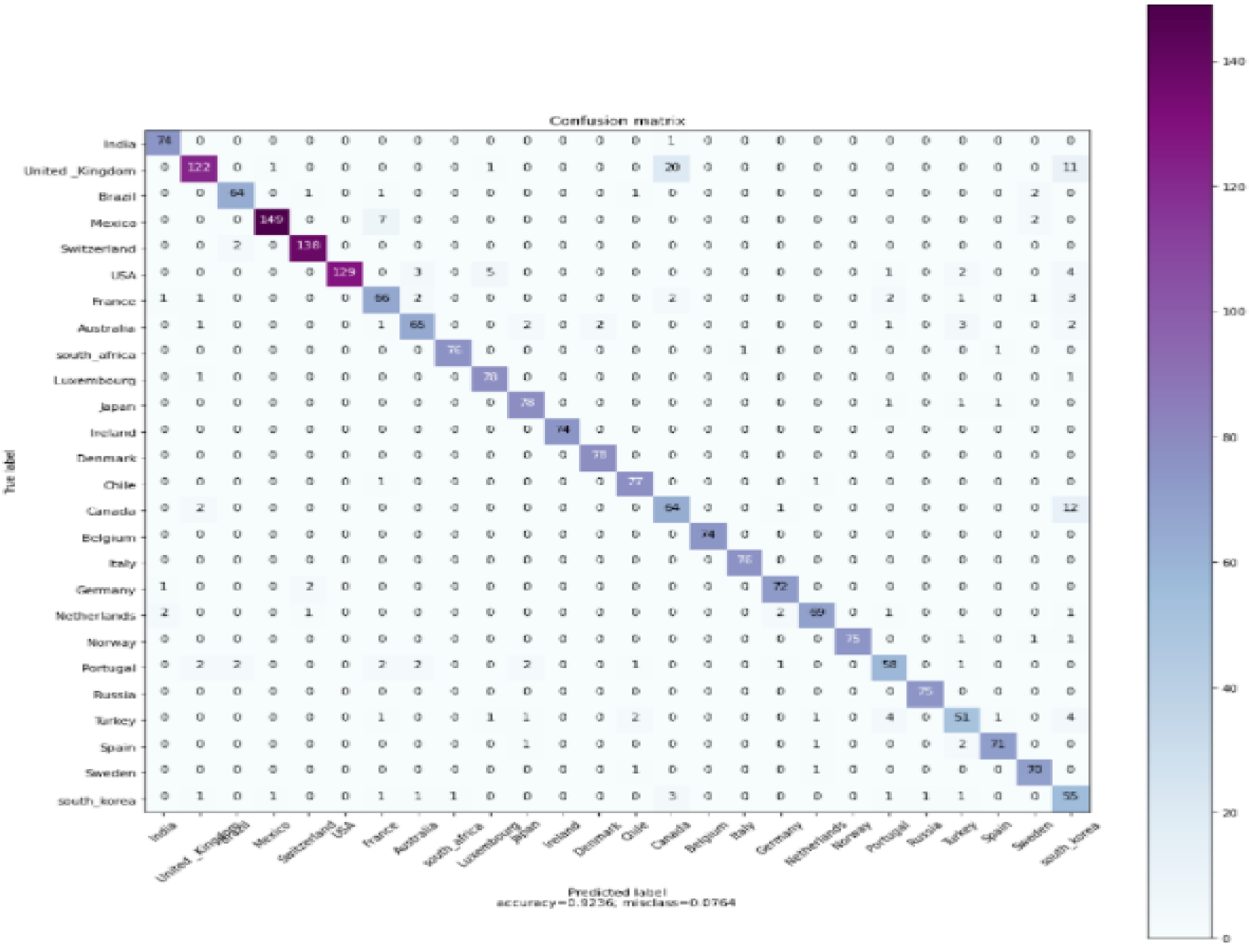


Figure 25: confusion matrix obtained from validation dataset of model 2.

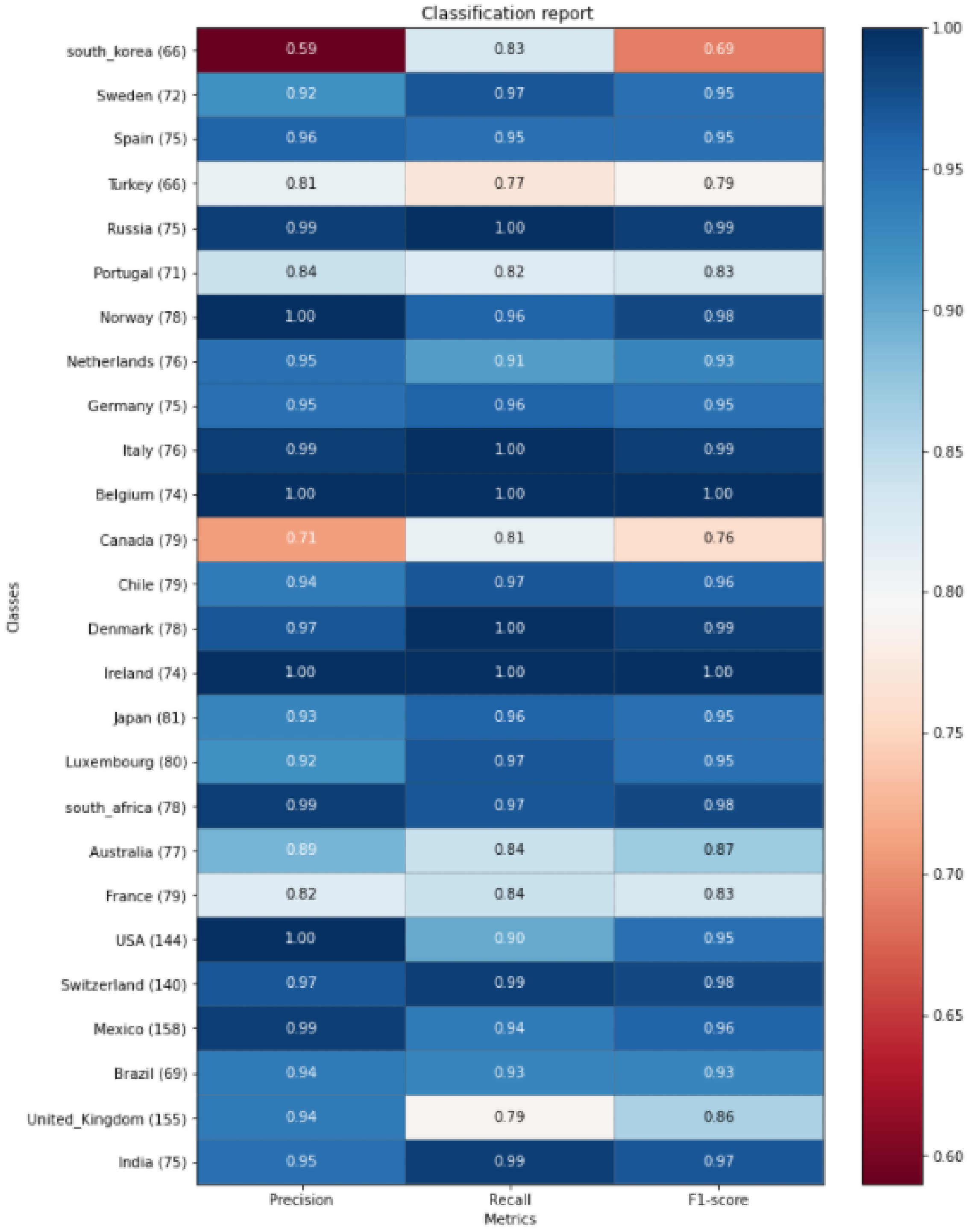


Figure 26: classification report obtained from validation dataset of model 2.

In the next section a discussion on the results achieved by the deep learning algorithms is provided and a comparison of the results found in this study with the most closely related study is presented. A technique that can be used to detect variants is described and the chapter concludes with an epilogue study that further validates the effectiveness.

Discussion

The variant classification model performance was outstanding at all evaluation metrics, and it was able of distinguishing between variants at high rate we believe the results of the model are due to many reasons; the availability of variants data and the balanced dataset used for the training but most of all the significant difference in genomes that is represented in viral mutations and evolution.

Similarly, the geographic classification model performed well overall, it was able to distinguish between genomes of different countries. however, the recall in (Figure 26) indicates there were misclassed sequences for example in the United Kingdom dataset there were 33 predictions out of 155 sequences that didn't match the actual class, means 122 are TP which can be observed in the diagonal of the confusion matrix (Figure 25), 20 out of 33 were predicted to be in Canada, 11 as south Korea's, and two other misclassed in Mexico and Luxembourg, the obtained results can be translated to high similarity between sequences in these countries in the dataset, further investigation is needed to improve the model and confirm the hypothesis.

A comparison was made to related works (Table 7), in terms of method, aim and results.

1st criteria Input data:

-Pangolin: user submits a multi sequence fasta file contains the genomes of sars-cov2.

- Covidex: user submits a multi sequence fasta file contains the genomes of sars-cov2.

- Vartrack: user submits a multi sequence fasta file contains the genomes of sars-cov2 but prediction is limited to 9 major variants and 26 countries.

2nd criteria Method:

-Pangolin: Alignment of the input genome against a set of pre-defined subtype reference sequences using mafft & Iq-tree, requires an expert to run.

- Covidex: Implementation of random forest trained over a k-mer database.

- Vartrack: implementation of neural networks trained over large k-mer database.
3rd criteria: Aim and Results:

-Pangolin: Pangolin assigns a global lineage to query SARS-CoV-2 genomes by estimating the most likely placement within a phylogenetic tree of representative sequences from all currently defined global SARS-CoV-2 lineages, it is accurate but computationally expensive.

- Covidex: classification of viral genomes in pre-defined clusters, it is fast with error rate less than 1.5 %.

- Vartrack: allows a user to assign a SARS-CoV-2 genome sequence the most likely lineage (Pango lineage) and geographic origin, fast with error rate less than 1% on lineage and less than 13% on geographic location.

Table 7: Comparison of VarTrack to other related work

Criteria Related work	Input data	Method	Aim and Results
Pangolin [13]	Multiple sequence fasta file	Alignment of the input genome against a set of pre-defined subtype reference sequences using mafft & Iq-tree.	Identify variants, accurate but computationally expensive
Covidex [14]	Multiple sequence fasta file	Implementation of random forest trained over a k-mer database	Identify variants, fast with error rate less than 1.5 %
Vartrack [15]	Multiple sequence fasta file	CNN model trained on large k-mer dataset	Identify variants and their geographic origin, fast with error rate less than 1% on lineage and less than 13% on geographic location.

Conclusion

Conclusion

In this paper we proposed a deep learning approach" VarTrack" to help investigate and predict new SARS-CoV-2 variants as they occur, the underlying principle is the switch from expert-based processes toward more and more automated data-driven and learned approaches.

The convolutional approach applied for extracting features from k-mer representation of input sequences boosts performance and increases accuracy and efficiency, on both models.

As described previously both models exceeded state of the art results with very short computation time. However, there are limits to deep learning that should be taken into consideration given the broad excitement for these new approaches: it requires large amounts of data that may not be available when working with experimental biological systems, it is limited in the capacity to discern mechanistic components, and can reflect biases and inaccuracies inherent in the data fed to them.

The black-box nature of deep learning models brings new challenges to biological applications. It is usually very difficult to interpret the output of a given model from a biological point of view, which limits the application of the model**.**

Since the convolutional neural network has an ability of extracting high-level abstraction features, we will study about extracted features from the model's convolutional layers to see if there are any interesting and meaningful features extracted to get more insights on the evolution of the virus and track mutations to help dealing with variants of interest and concern.

We will provide analytical web application that targets a wide population of scientists with a classification platform that can be easily extended to other viruses, dashboards and world map to enhance visibility and summarize the distribution of viruses geographically

We will apply the models on other sequence data to solve other problems in genomics, the key issues are genome classification and sequence annotation (e.g; motif extraction, gene functions)

REFERENCES

REFERENCES

1. Alexandra C. Walls, Young-Jun Park, M. Alejandra Tortorici, Abigail Wall, Andrew T. McGuire, and David Veesler. (2020) ‘Structure, Function, and Antigenicity of the SARS-CoV-2 Spike Glycoprotein’, Cell, 181(2), pp. 281-292.e6. doi: 10.1016/j.cell.2020.02.058.
2. Franziska Hufsky, Kevin Lamkiewicz, Alexandre Almeida, Abdel Aouacheria, Cecilia Arighi, Alex Bateman. et al. (2021) ‘Computational strategies to combat COVID-19: useful tools to accelerate SARS-CoV-2 and coronavirus research’, Briefings in Bioinformatics, 22(2), pp. 642–663. doi: 10.1093/bib/bbaa232.
3. David M. Morens, Joel G. Breman, Charles H. Calisher, Peter C. Doherty, Beatrice H. Hahn, Gerald T. Keusch et al. (2020) ‘The Origin of COVID-19 and Why It Matters’, The American Journal of Tropical Medicine and Hygiene, 103(3), pp. 955–959. doi: 10.4269/ajtmh.20-0849.
4. Mei-Yue Wang, Rong Zhao, Li-Juan Gao, Xue-Fei Gao, De-Ping Wang and Ji-Min Cao. (2020) ‘SARS-CoV-2: Structure, Biology, and Structure-Based Therapeutics Development’, Frontiers in Cellular and Infection Microbiology, 10. doi: 10.3389/fcimb.2020.587269.
5. M. Saqib Nawaz, Philippe Fournier-Viger, Abbas Shojaee, Hamido Fujita (2021) ‘Using artificial intelligence techniques for COVID-19 genome analysis’, Applied Intelligence, 51(5), pp. 3086–3103. doi: 10.1007/s10489-021-02193-w.
6. Michail Galanopoulos, Aris Doukatas, and Maria Gazouli. (2020) ‘Origin and genomic characteristics of SARS-CoV-2 and its interaction with angiotensin converting enzyme type 2 receptors, focusing on the gastrointestinal tract’, World Journal of Gastroenterology, 26(41), pp. 6335–6345. doi: 10.3748/wjg.v26.i41.6335.
7. Genomic sequencing of SARS-CoV-2: a guide to implementation for maximum impact on public health. Available at: https://www.who.int/publications-detail-redirect/9789240018440 (Accessed: 23 June 2021).
8. Sapkota, A. (2020) Structure and Genome of SARS-CoV-2 (COVID-19) with diagram, Microbe Notes. Available at: https://microbenotes.com/structure-and-genome-of-sars-cov-2/ (Accessed: 21 June 2021).
9. Canada, P. H. A. of (2021) SARS-CoV-2 variants: National definitions, classifications and public health actions. Available at: https://www.canada.ca/en/public-health/services/diseases/2019-novel-coronavirus-infection/health-professionals/testing-

diagnosing-case-reporting/sars-cov-2-variants-national-definitions-classifications-public-health-actions.html (Accessed: 23 June 2021).

10. CDC (2020) Coronavirus Disease 2019 (COVID-19), Centers for Disease Control and Prevention. Available at: https://www.cdc.gov/coronavirus/2019-ncov/variants/variant-info.html (Accessed: 23 June 2021).

11. FDA reports on viral mutations and SARS-CoV-2 tests (2021) Healthcare Purchasing News. Available at: https://www.hpnonline.com/infection-prevention/screening-surveillance/article/21217625/fda-reports-on-viral-mutations-and-sarscov2-tests (Accessed: 14 July 2021).

12. Courtney E. French, Isabelle Delon, Helen Dolling, Alba Sanchis-Juan, Olga Shamardina, Karyn Mégy. et al. (2019) 'Whole genome sequencing reveals that genetic conditions are frequent in intensively ill children', Intensive Care Medicine, 45(5), pp. 627–636. doi: 10.1007/s00134-019-05552-x.

13. Áine O'Toole, Emily Scher, Anthony Underwood, Ben Jackson, Verity Hill, John T. McCrone. et al. (2021) 'Assignment of Epidemiological Lineages in an Emerging Pandemic Using the Pangolin Tool', Virus Evolution, p. veab064. doi: 10.1093/ve/veab064.

14. BEAST 2. Available at: https://www.beast2.org/ (Accessed: 14 July 2021).

15. Marco Cacciabue, Pablo Aguilera, María Inés Gismondi, and Oscar Taboga. (2020) 'Covidex: an ultrafast and accurate tool for virus subtyping', bioRxiv, p. 2020.08.21.261347. doi: 10.1101/2020.08.21.261347.

16. Big Data: Astronomical or Genomical?. Available at: https://journals.plos.org/plosbiology/article?id=10.1371/journal.pbio.1002195 (Accessed: 14 July 2021).

17. Heath, N. What is AI? Everything you need to know about Artificial Intelligence, ZDNet. Available at: https://www.zdnet.com/article/what-is-ai-everything-you-need-to-know-about-artificial-intelligence/ (Accessed: 23 June 2021).

18. Advani, V. (2021) M, GreatLearning Blog: Free Resources what Matters to shape your Career! Available at: https://www.mygreatlearning.com/blog/what-is-artificial-intelligence/ (Accessed: 14 July 2021).

19. Bheemaiah, K., Esposito, M. and Tse, T. What is machine learning?, The Conversation. Available at: http://theconversation.com/what-is-machine-learning-76759 (Accessed: 14 July 2021).

20. Supervised Learning - an overview | ScienceDirect Topics. Available at: https://www.sciencedirect.com/topics/computer-science/supervised-learning (Accessed: 14 July 2021).

21. Upadhyay, A. (2020) Classification In Machine Learning, Medium. Available at: https://medium.com/analytics-vidhya/classification-in-machine-learning-ed30753d9461 (Accessed: 14 July 2021).

22. ‘Regression vs Classification in Machine Learning: What is The Difference? | Springboard Blog’. Available at: https://in.springboard.com/blog/regression-vs-classification-in-machine-learning/ (Accessed: 14 July 2021).

23. Bhadra, R. (2019) What is a Neural Network?, Medium. Available at: https://towardsdatascience.com/what-is-a-neural-network-a02b3c2fe3fa (Accessed: 14 July 2021).

24. Husssein, A. (2020) Hisaack/Artificial-Neural-Networks. Available at: https://github.com/Hisaack/Artificial-Neural-Networks (Accessed: 14 July 2021).

25. ‘Deep Learning for Student Competitions’ Student Lounge. Available at: https://blogs.mathworks.com/student-lounge/2019/05/29/deep-learning-for-student-competitions/ (Accessed: 14 July 2021).

26. Machine Learning &amp; AI Panel Discussion. Available at: https://www.linkedin.com/pulse/machine-learning-ai-panel-discussion-vivek-kumar-1 (Accessed: 14 July 2021).

27. Sarker, I. H. (2021) ‘Machine Learning: Algorithms, Real-World Applications and Research Directions’, SN Computer Science, 2(3), p. 160. doi: 10.1007/s42979-021-00592-x.

28.

29. Biological Vs Artificial Neural Network, Jobs EcityWorks (no date). Available at: https://www.ecityworks.com/biological-vs-artificial-neural-network (Accessed: 14 July 2021).

30. What Is The Relation Between Artificial And Biological Neuron? | 码农网 (no date). Available at: https://www.codercto.com/a/113149.html (Accessed: 14 July 2021).

31. SHARMA, S. (2019) What the Hell is Perceptron?, Medium. Available at: https://towardsdatascience.com/what-the-hell-is-perceptron-626217814f53 (Accessed: 14 July 2021).

32. Manaswi, N. K. (2020) Generative Adversarial Networks with Industrial Use Cases: Learning How to Build GAN Applications for Retail, Healthcare, Telecom, Media, Education, and HRTech. BPB Publications.

33. Nejad, A. (2021) Convolutional Neural Network Champions —Part 1: LeNet-5, Medium. Available at: https://towardsdatascience.com/convolutional-neural-network-champions-part-1-lenet-5-7a8d6eb98df6 (Accessed: 14 July 2021).

34. Complete Natural Language Processing Guide For Beginners In 2021 - Buggy Programmer (no date). Available at: https://buggyprogrammer.com/what-is-natural-language-processing/ (Accessed: 14 July 2021).

35. Deep Learning Vs NLP: Difference Between Deep Learning & NLP (2020) upGrad blog. Available at: https://www.upgrad.com/blog/deep-learning-vs-nlp/ (Accessed: 14 July 2021).

36. Why Deep Learning Is a Perfect Match for Natural Language Processing ｜Appier (2019) Appier. Available at: https://www.appier.com/blog/why-deep-learning-is-a-perfect-match-for-natural-language-processing/ (Accessed: 14 July 2021).

37. Shen, D. (2009) 'Text Categorization', in LIU, L. and ÖZSU, M. T. (eds) Encyclopedia of Database Systems. Boston, MA: Springer US, pp. 3041–3044. doi: 10.1007/978-0-387-39940-9_414.

38. Machine Learning Approach to Search Query Classification: Computer Science & IT Book Chapter | IGI Global (no date). Available at: https://www.igi-global.com/chapter/machine-learning-approach-search-query/22008 (Accessed: 14 July 2021).

39. Amazon Comprehend Medical, Amazon Web Services, Inc. Available at: https://aws.amazon.com/fr/comprehend/medical/ (Accessed: 23 June 2021).

40. Sentiment Analysis – The Go-To Guide (no date) MonkeyLearn. Available at: https://monkeylearn.com/sentiment-analysis/ (Accessed: 23 June 2021).

41. .Peters, J. (2019) Google is bringing spam detection and verified business messaging to Messages, The Verge. Available at: https://www.theverge.com/2019/12/12/21012601/google-messages-spam-detection-verified-sms-features (Accessed: 23 June 2021).

42. Detecting fake news at its source (no date) MIT News | Massachusetts Institute of Technology. Available at: https://news.mit.edu/2018/mit-csail-machine-learning-system-detects-fake-news-from-source-1004 (Accessed: 23 June 2021).